

ISD61S00 Evaluation System

User's Manual



1.	Getting Started	4
1.1	Install the ISD61S00 Evaluation System	4
2.	Evaluation Board Overview.....	7
2.1	USB Controller and Power Sources	7
2.2	ISD61S00 and Related Circuitry	8
2.3	Clare DAA.....	9
2.4	Discrete DAA.....	10
3.	ISD61S00 Software Utility Overview	12
3.1	PCM & PLL	12
3.2	Voice Prompt.....	14
3.2.1	Keystrokes to Re-order VPs.....	15
3.3	Reserved Memory	15
3.4	Voice Macro	16
3.5	Path Config.....	18
3.6	Memory Map.....	20
3.7	Archive and Clean.....	22
3.8	DTMF Detect	23
3.9	Tone/DTMF Gen.....	25
3.10	FSK Detect	27
3.11	FSK Generation	29
3.12	Register Map.....	31
3.13	Analog	32
3.14	Demo	36
3.15	AEC/LEC.....	38
3.16	AGC.....	45
3.17	GPIO/INT	48
3.17.1	GPIO	48
3.17.2	Interrupt.....	49
3.18	Detect.....	50
3.18.1	CAS detect	51
3.18.2	ATD	52
3.18.3	Ring Detect	54
3.18.4	CPT Detect.....	55
3.18.5	VD.....	56
3.18.6	Demo.....	57
4.	Evaluation Board Control	59
4.1	Audio Cmds.....	59
4.2	Digital Cmds	61
5.	Board Schematic	64
5.1	USB Controller and Power Sources	64
5.2	ISD61S00 and Related Circuitry.....	65
5.3	Clare DAA.....	66

5.4	Discrete DAA.....	67
6.	Revision History	68

1. Getting Started

The ISD61S00 is a feature chip for the security and telephony industry. The ISD61S00 evaluation system is an integrated tool kit which enables users to evaluate the features of ISD61S00 including: a powerful macro scripting ability to facilitate audio prompting in a multi-language environment and simple address-free recording and playback; and telephony based data communications including DTMF detection and generation, FSK modem functions from 75-1200 baud, CAS and CPT (Call Progress Tone) detection and ring detection.

1.1 Install the ISD61S00 Evaluation System

Please follow the steps below to set up the evaluation system:

1. Install the ISD61S00 software utility before plugging the EV board onto PC.
2. After the ISD61S00 software utility is properly installed, plug the EV board onto PC.
 1. Users shall see a message popped up; select 'No, not this time', and then click 'Next'.



Figure 1-1 Install USB Driver - 1

2. Select 'install from a list or specific location (Advanced)', and then click 'Next'.

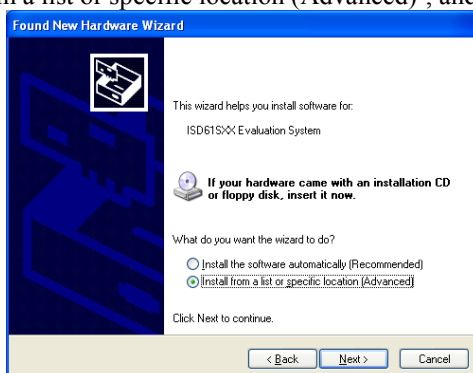


Figure 1-2 Install USB Driver - 2

3. Click 'Browse'.

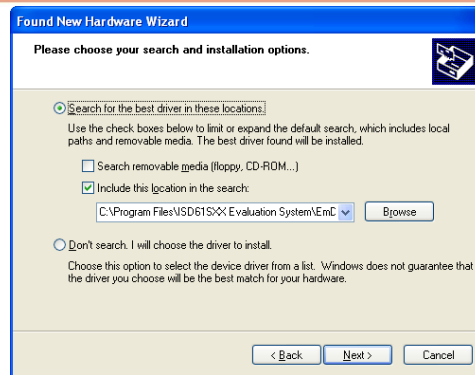


Figure 1-3 Install USB Driver - 3

4. USB driver is located in the ISD61S00 software utility installation folder, which by default is "c:\Program Files\ISD61SXX Evaluation System\Driver"; Click 'OK'.



Figure 1-4 Install USB Driver - 4

5. Users shall then see the installing of USB driver in progress.

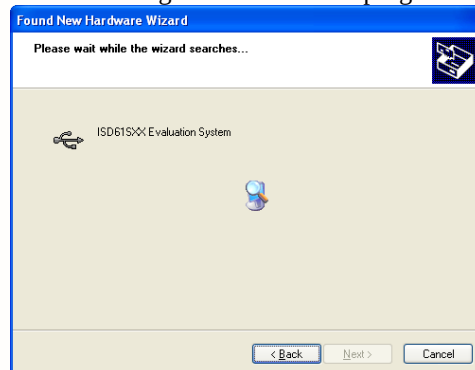


Figure 1-5 Install USB Driver - 5

6. After the installation of USB driver, reset the USB by unplugging and plugging the board.

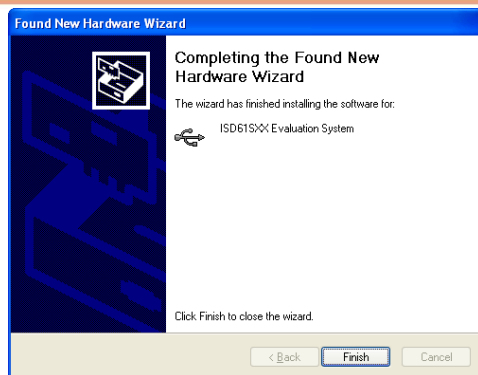


Figure 1-6 Install USB Driver - 6

2. Evaluation Board Overview

Figure 2-1 below is the ISD61S00 Evaluation board. The board has 4 main areas:

1. USB Controller and Power Sources
2. ISD61S00 and Related Circuitry
3. Clare DAA
4. Discrete DAA

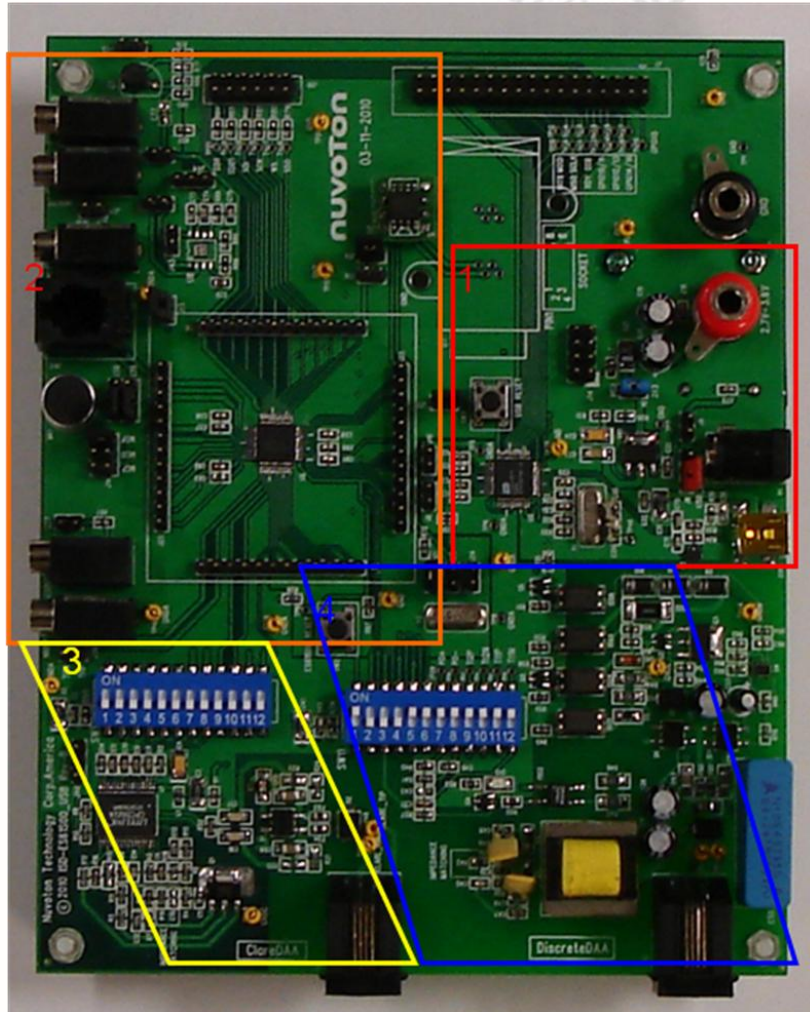


Figure 2-1 ISD61S00 Evaluation Board

2.1 USB Controller and Power Sources

The board could be powered by USB power (A) or an external 5V power (B), as determined by the jumper J6 (C) or 3 AAA batteries by the jumper J9 (?). VR1 (D) is a 5V-to-3V regulator to provide the needed 3V for the ISD61S00. Or, users could also power the board by a 3V power source via the on-board banana jack (E); the jumper J5 (F) must be removed if this is the case.

The evaluation system is controlled by an on-board USB controller W681308 (G). Users can capture the controlling signals between the USB controller and the ISD61S00 on the on-board headers (H). To reset the USB controller, users could either press the RESB button (I) or unplug and plug the USB cable (A). The on-board USB controller has its own microphone (J) and speaker (K) interfaces.

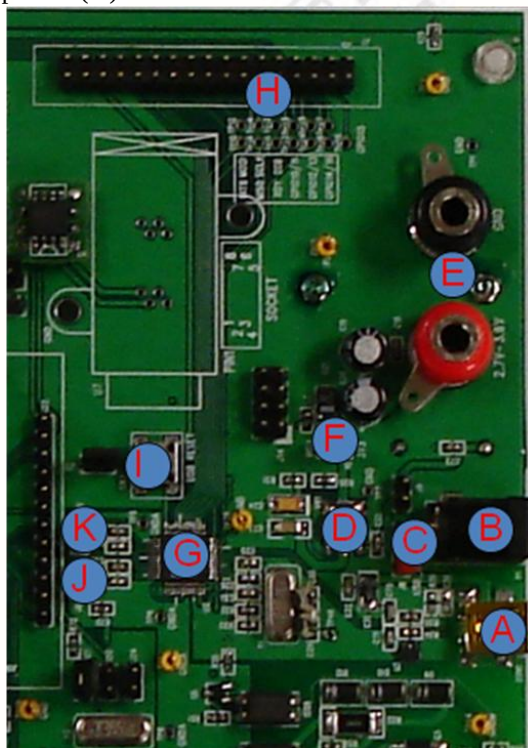


Figure 2-2 USB Controller and Power Sources

2.2 ISD61S00 and Related Circuitry

ISD61S00 (A) is clocked by the crystal or the external clock source. There is a 12.288MHz crystal (B) on the board. The jumper J21 (C) must be removed if users want to clock the ISD61S00 by an external clock source.

ISD61S00 is a ROM-less part, which uses external serial flash as the storage media. On board there are a DIP-8 socket (E) and a flash socket footprint (F, socket not included); users can select either one by the jumpers J29 (G) and J30 (H). For the DIP-8 socket, flash memory will be soldered on an SOIC adaptor board so that users can replace the flash easier. For the flash socket, users can turn the board into a flash programmer by soldering the flash socket.

Around the ISD61S00 are the related connectors and circuitries:

- I2S interface (I) shares the GPIO pins 4 ~9; by default these pins are pull-up input pins; to set them as I2S users have to set the corresponding PORT_AF (port alternate function) registers.
- J31 and JAUX (J) are the pin-header and jack for Aux-out. Aux-out is a single-ended voltage output, which needs an external amplifier to drive the speaker.
- J25 and JAUXSPK (S) are the pin-header and jack for Aux-out amplified. The ISD8101 U10 takes the single end voltage as input and drives the speaker with differential voltage by JAUXSPK.

- If user wants to drive speaker by external amplifier ISD8101, user may put jumpers on J43 and J44 2-3 to turn on the ISD8101.
- J27 and 3.5mm Jack (**K**) are the pin-header and jack for the device speaker driver outputs.
- MK1 (**L**) is the on-board microphone, which is configured as a single-ended microphone input. To use the microphone, user can put jumper on J19 (**N**) 5-6, J38 1-2 and J39 1-2 (**R**).
 - If users want to record messages via Aux-in, users can put jumpers on J19 (**N**) 1-2 & 3-4, remove the jumper on J19 (**N**) 5-6, and then record through the jack phone connector JADCIN (**O**).
- J36 and JTI11N (**P**) are the pin-header the jack connector of the TI1P/TI1N.
- J14 is the keypad connector for standalone operation. The keypad consists of
 - A 4 x 4 key pad.
 - A uC to talk to the main board.
 - A display for Caller ID.

To reset the ISD61S00, users can press the RESET button (**Q**)

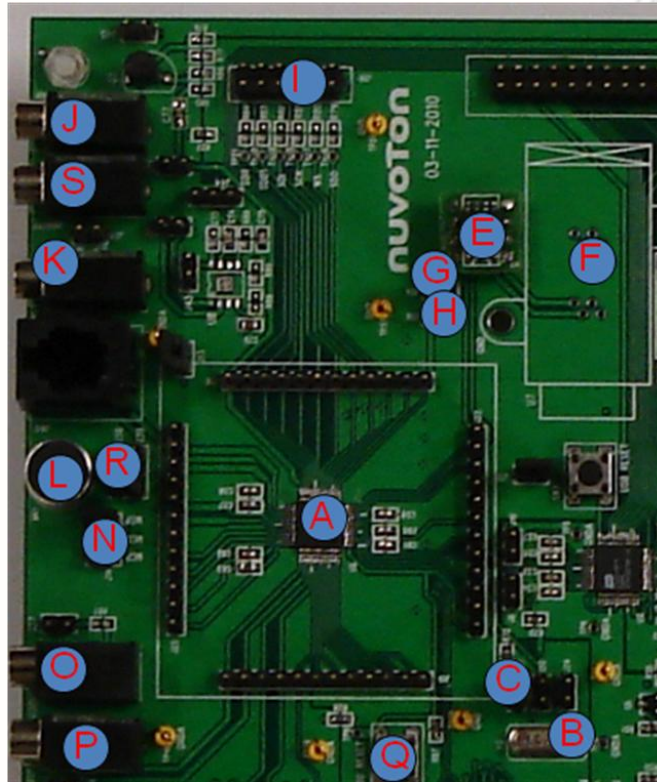


Figure 2-3 ISD61S00 and Related Circuitry

2.3 Clare DAA

With the Clare DAA configuration, the phone signal (**A**) goes to the on-board Clare CPC5622 (**B**) to talk to the ISD61S00. The Clare CPC5622 (**B**) converts the 2-wire phone signal (TIP and RING) to 4 wires (TI and PO) and processes the controlling signals including the Hook Control, Hook Detection, Loop Detection, Ring Detection, and Pulse Dial. This simple configuration reduces the number of discrete components as in the Discrete DAA.

To use the Clare DAA, users need to switch SW1 (C) as in Figure 2-5. The switch SW1 (C) of Clare DAA is a counterpart of the switch SW11 of Discrete DAA.

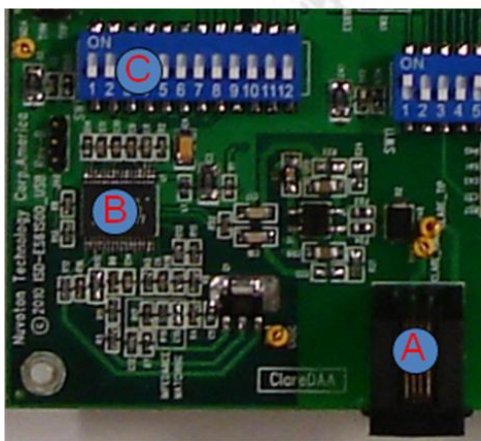


Figure 2-4 Clare DAA

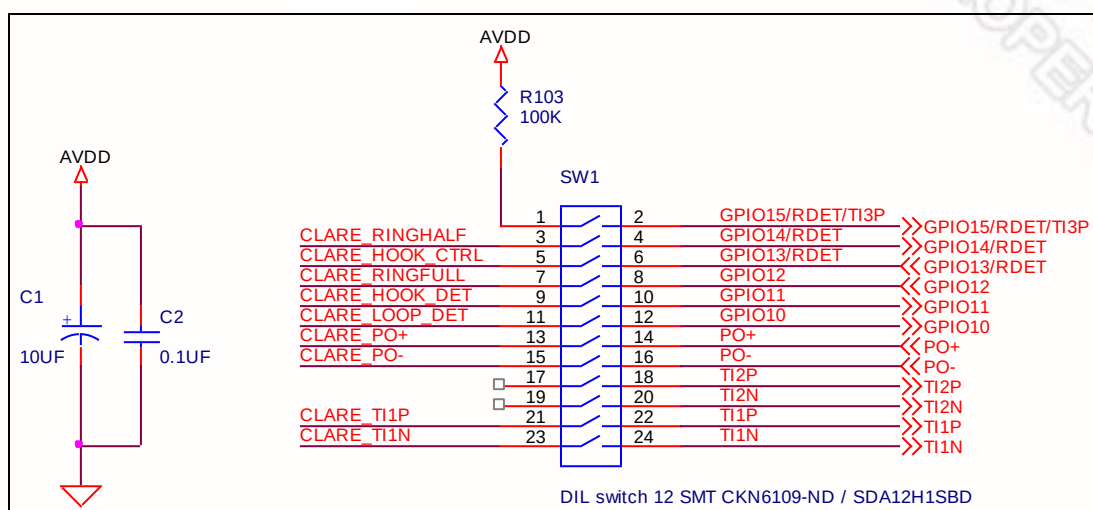


Figure 2-5 Switch of Clare DAA

2.4 Discrete DAA

Unlike the Clare DAA, the Discrete DAA configuration uses discrete components to convert the phone line (A) and process the controlling signals including the Hook Control, Hook Detection, Loop Detection, Ring Detection, and Pulse Dial. For details of Discrete DAA configuration, please refer to section 5.4 *Discrete DAA*.

To use the Discrete DAA, users need to switch SW11 (B). The switch SW11 (B) of Discrete DAA is a counterpart of the switch SW1 of Clare DAA.

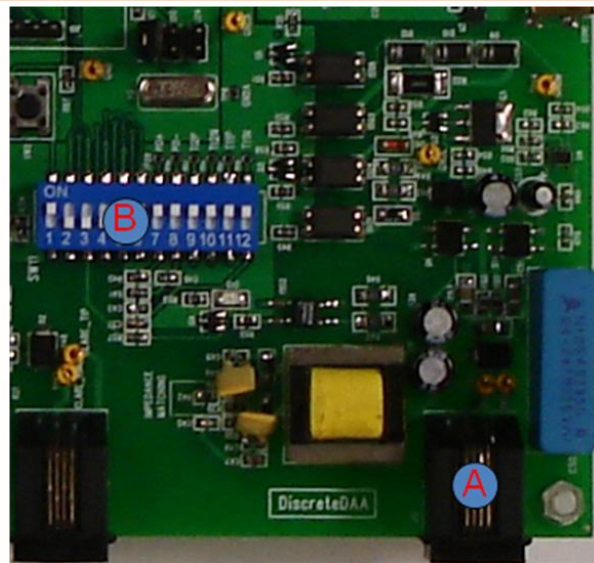


Figure 2-6 Discrete DAA

3. ISD61S00 Software Utility Overview

There is an example project named I61SXX_Sample. This section introduces the ISD61S00 software utility panel by panel based on this example project.

To open the example project: launch the ISD61S00 software utility. On the menu bar, click 'Project' -> 'Open'; users can find the example project in "c:\Program Files\ISD61SXX Evaluation System\I61SXX_Sample".

There are three subfolders in the project folder:

- BinFiles: where the compressed Voice Prompts are stored.
- WavFiles: where the re-sampled and de-compressed Voice Prompts are stored.
- WavSrc: where wave sources of Voice Prompts are stored.

3.1 PCM & PLL

Users configure the Digital Audio Interface (PCM/I2S) and the Clock Generation on this panel.

ISD61S00 has a digital audio interface consisting of two physical transitive/receive channels; both channels share common clock and synchronization. With digital audio, ISD61S00 can be configured as master (ISD61S00 generates clock and frame sync) or slave (ISD61S00 accepts clock and frame sync). The digital audio interface has two modes;

- In I2S mode, PCM interface conforms to the I2S digital audio interface convention.
- In PCM mode, data transfer is synchronized to the rising edge of the frame sync signal.

For register details please refer to the design guide, section 8.4 Digital Audio Interface Control.

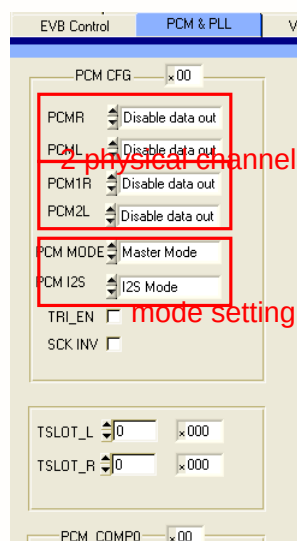


Figure 3-1 PCM/I2S Control

Figure 3-2 below shows the block diagram of the clock generation. The PLL is mainly used to multiply the external input clock by a fractional number (the $f_2 = R * f_1$ in the diagram). The dividers before and after the PLL are optional for users to get the frequency within a certain range. The target IMCLK is 24.576MHz.

For register details please refer to the design guide, section 8.3 Clock Generation and PLL.

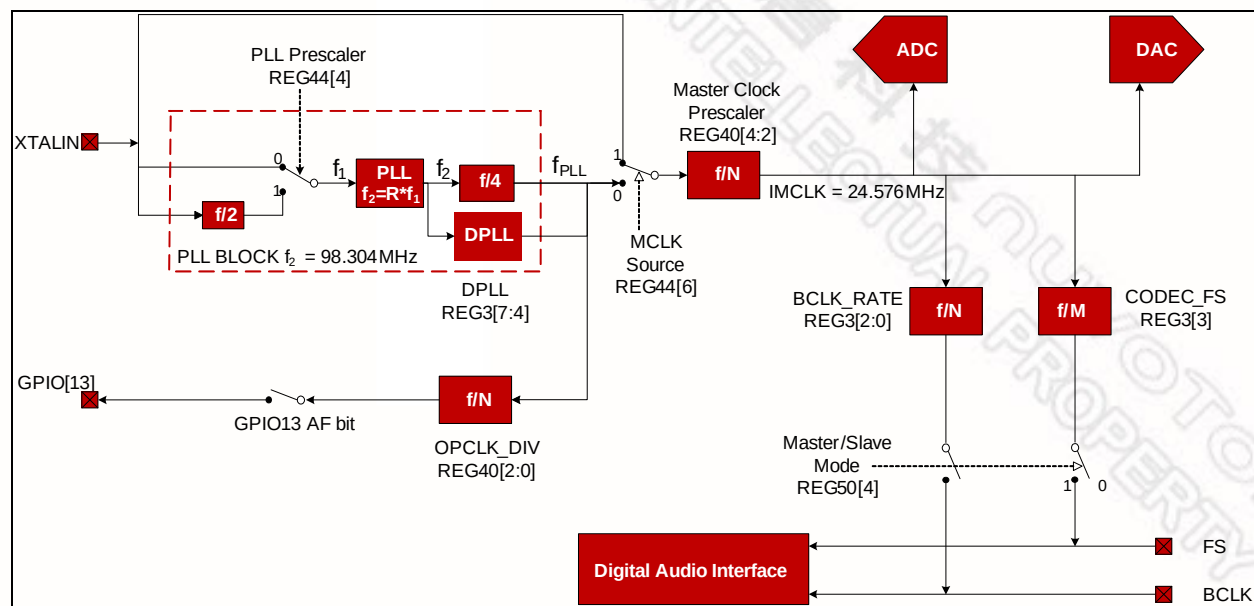


Figure 3-2 Clock Generation and PLL

Figure 3-3 Clock and PLL Setting

3.2 Voice Prompt

Voice prompts are pre-recorded audio of any length, from short words, phrases or sound effects to long passages of music. These Voice Prompts can be played back in any order as determined by the users and applications. A voice prompt consists of two components:

- (1) An index pointing to the pre-recorded audio.
- (2) Pre-recorded audio.

Before adding voice prompts, first set the default sampling rate (A) and compression algorithm (B).

To add voice prompts, follow the steps below:

- (1) Set sampling rate (C) and compression algorithm (D) for the VPs you are going to add. Users can change the individual VP's sampling rate (E) and/or compression algorithm (F) later when evaluating the quality.
- (2) Add voice prompts (G).
- (3) Repeat steps (1) & (2).
- (4) Change the VP tags (H) if needed; by default, the tags are the filenames.
- (5) Re-order the VPs if needed; select VP by clicking on the square (I), and click on the up/down arrows (J) to re-order the VPs.

If users change the clock configuration, they must re-compress all VPs (K) because clock setting is part of the VP compression.

Users can evaluate the VPs through PC or EV board:

- Through PC (L):
 - i. Original voice prompts before re-sampling & compression.
 - ii. Re-sampled but not yet compressed voice prompts.
 - iii. Compressed voice prompts.
- Through EV board:
 1. Plug in the EV board and the 'Play to device' button (M) shows up.

The added voice prompts will appear on the 'Voice Macro' screen for voice macro script programming.

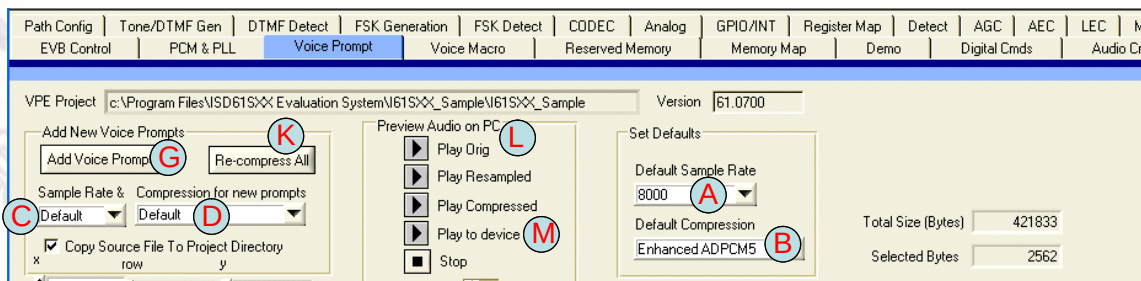


Figure 3-4 Voice Prompt screen

3.2.1 Keystrokes to Re-order VPs

Besides the up/down arrows (J), users can also use the keystrokes to re-order the VPs:

- Select VP by clicking on the square (I)
- Press M on keyboard and a pop-up box shows up.
- Enter the new VP # to re-order the VP.

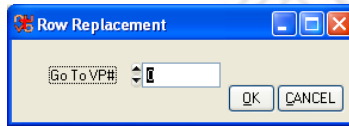


Figure 3-5 Re-order VPs

3.3 Reserved Memory

Recording Memory Pointer (RMP) divides the ISD61S00 memory space into two blocks:

- Reserved Memory,
- Recording Memory.

RMP is a two-byte address pointer pointing to a 4kbyte memory sector which is the first sector available for message recordings. Memory between address zero and the RMP pointer is considered the Reserved Memory for pre-recorded audio (Voice Prompts), pre-programmed macro scripts (Voice Macros), digital read/write access for other applications (User Data), and memory sectors reserved for the first sector of the reserved messages (Reserved Sectors). The memory between the RMP and the end of memory is considered the Recording Memory allocated for recording messages (Message Recordings).

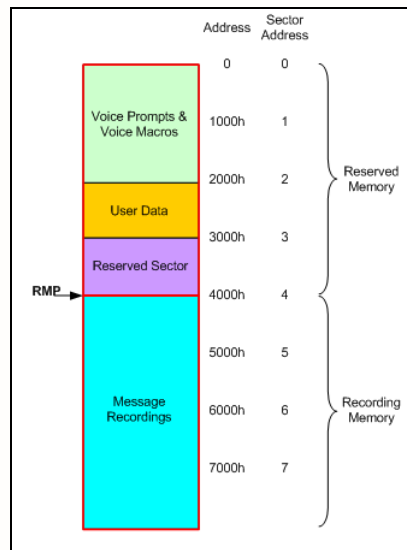


Figure 3-6 Example of ISD61S00 Memory Map

On the 'Reserved Memory' screen, users reserve memory sectors for message recordings:

- Empty messages (A):
 - An Empty Message is an empty 4kByte memory sector reserved for message recordings. An SPI record command pointing to the Empty Message starts the message recording and will continue to

record to the free memory sectors located in Recording Memory if the message recording is over 4kByte.

- Re-recordable messages (B):
 - Users can have a default message when reserving the memory sectors for a message recording. For a re-recordable message, the first sector will be allocated in 'Reserved Sectors'; the rest of the message will be allocated in 'Recording Memory'.

Please note only ADPCM, (D)uLaw, and PCM compressions are supported for reserved messages, not Enhanced-ADPCM or VBR.

To record (or playback) a message, the microcontroller simply issues a record (or playback) command with starting address pointing to the reserved initial address. The reserved messages are also listed on the 'Voice Macro' screen for voice macro script programming.

Memory sectors can also be assigned in Reserved Memory for digital read/write access for other application tasks (User Data). To reserve empty memory sectors, click the button 'Add Reserved Memory' (C) and then adjust the 'Number of Sectors' (D). To reserve (import) existing data, click the button 'Add Data' (E); the file size will be shown on 'number of sectors' (F).

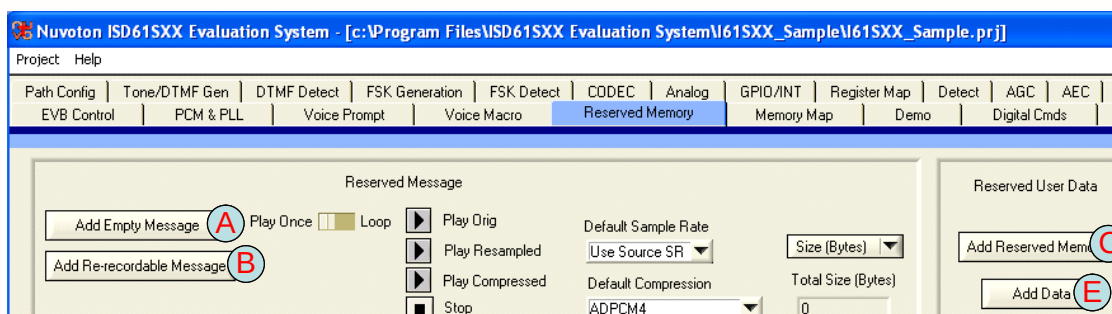


Figure 3-7 Reserved Memory

3.4 Voice Macro

Voice Macro is a powerful scripting feature integrated into the ISD61S00. Voice Macro editor enables users to create scripted commands to play voice prompts (A), play reserved messages (B), insert silence (C), power down the ISD61S00 (D), and configure the ISD61S00 (E) such as setting play or record paths. As mentioned earlier, the voice prompts (A) are from the 'Voice Prompt' panel, and the reserved messages (B) are from the 'Reserved Memory' panel, respectively. Each VM is assigned a tag and an index, where it can be executed with a single SPI command referring to the index.

There are 3 special Voice Macros:

- Index 0: VM0, Power-On Initialization (POI).
 - Executed automatically when power supplied or when a high pulse applied on RESET pin.
- Index 1: VM1, Power-Up (PU).

- Executed automatically when received power-up command.
- Index 2: VM2, Power-Up (WAKEUP).
 - Executed automatically when woke up from a GPIO interrupt.

POI, PU, and WAKEUP Macros allow users to customize how the part emerges from power-on, power-up, and wake-up conditions.

There are two special VM commands: FINISH (F) and WAIT INT (G).

- FINISH (F):
 - Finish the voice macro and exit. Voice Macros must be ended in either FINISH or PD (power down).
- WAIT INT (G):
 - Wait until current play command finishes before executing next macro instruction.

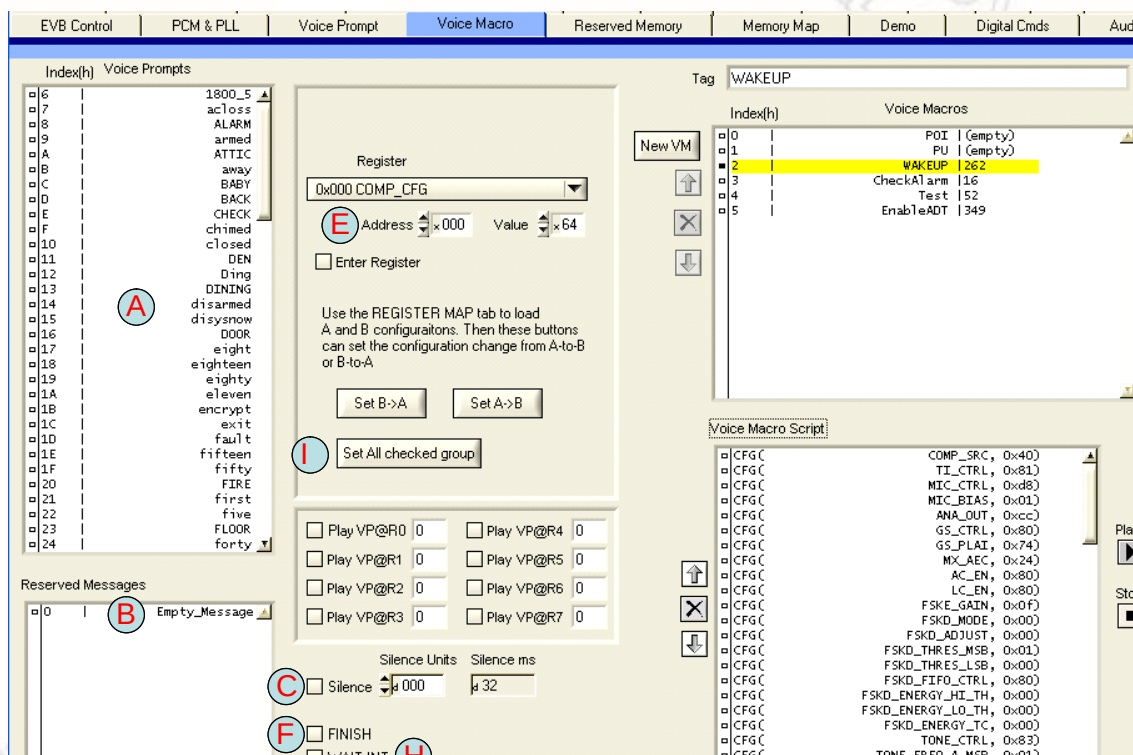


Figure 3-8 Voice Macro

The ISD61S00 has a built-in command buffer so that consecutive commands can be executed one after another seamlessly. However, some commands won't go through command buffer; therefore, they won't wait. Users may refer to the design guide; section 10 SPI Commands, Table 10-2, for details. In table 10-2 users may see that some commands can be executed regardless the status bit-1 CBUF_FULL (command buffer full). WAIT_INT (H) is used whenever a command in a voice macro intends to wait to be executed until previous play command finishes. A typical example will be adjusting volume in a voice macro. Commands for adjusting volume won't go through the command buffer; therefore, they will be executed immediately. Below is an example where the volume is immediately decreased when VP_A starts to play.

- Play VP_A
- Decrease Volume

- Play VP_B
- Finish.

If users want to decrease volume after VP_A finishes, the voice macro should be modified as below:

- Play VP_A
- WAIT_INT
- Decrease Volume
- Play VP_B
- Finish.

Please note that the commands in a Voice Macro are executed sequentially, and all voice macros shall be ended with PD or FINISH to finish the voice macro. Users shall see a warning message if the software detects a voice macro not ended with PD or FINISH.

3.5 Path Config

This panel helps users to configure the ISD61S00 Signal Path. The path diagram has 4 main parts:

1. Air input/output (microphone and speaker)
2. Phone-line input/output, and the DTMF/FSK generation and detection.
3. Mixers
4. Compression and de-compression

Figure 3-9 below shows a typical phone signal path:

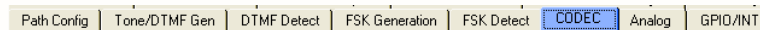
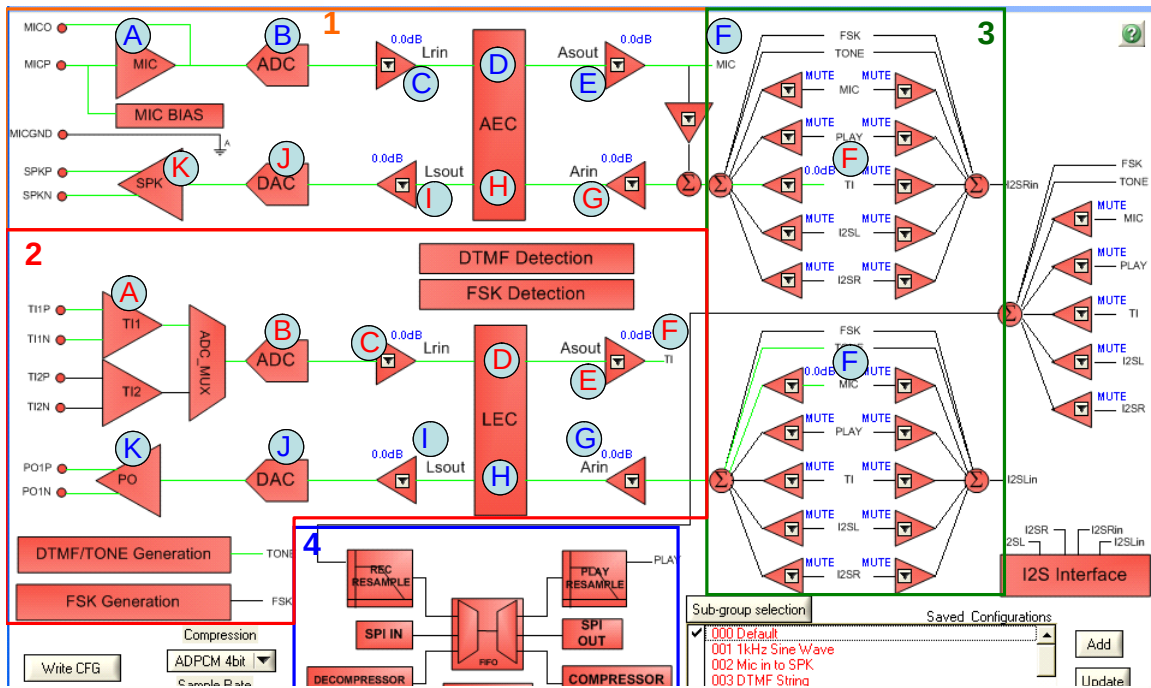
- This site to the other site:
 - Microphone (A) -> ADC (B) -> Lrin (C) -> AEC (D) -> Asout (E) -> MIC (F) -> Arin (G) -> LEC (H) -> Lsout (I) -> DAC (J) -> PO (K)
- The other site to this site:
 - TI1 (A) -> ADC (B) -> Lrin (C) -> LEC (D) -> Asout (E) -> TI (F) -> Arin (G) -> AEC (H) -> Lsout (I) -> DAC (J) -> SPK (K)

To enable or disable a path portion, users simply click on the related function block, and the software automatically switches to the corresponding panel. For example, to disable the ADC (B), users just click on the ADC block, and then the software will switch to the CODEC panel as in Figure 3-10. Users then uncheck the Air CODEC (L) to disable the ADC. (Please note that ADC and DAC share the hardware circuitry, so the ADC and DAC will be turned on or off at the same time).

Similarly, to tune the AEC (D) users just click on the AEC block, and the software utility automatically switches to the AEC panel as in Figure 3-11. For register details of AEC and LEC, please refer to design guide, section 8.18 Air and Line CODEC.

To change the of gain of Lrin (C), Asout (E), MIC (F) ... etc, users just click on the function block and then a gain setting bar shows up as the Figure 3-12.

The software utility has built in some default path settings as in Figure 3-13 for users' reference. To try a setting, users simply double-click on it. Or, users can click on it, modify the setting as needed, and then click the 'Write CFG'.



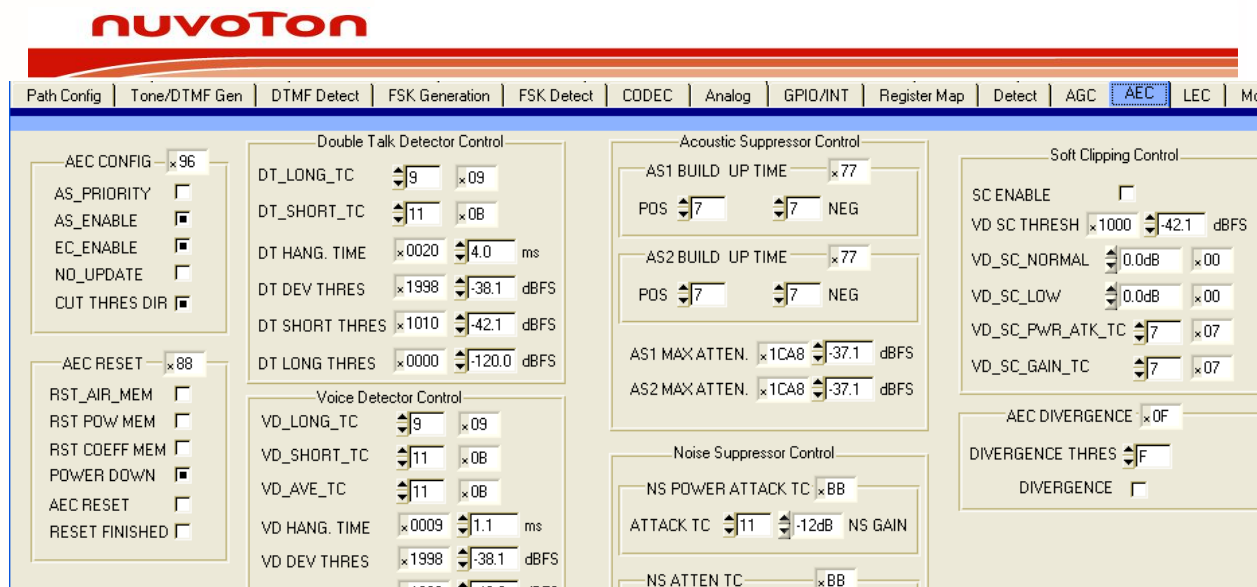


Figure 3-11 AEC Parameters

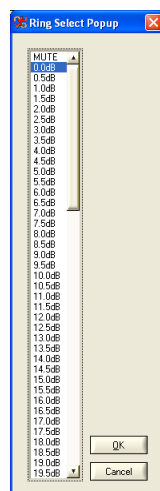


Figure 3-12 Gain Setting

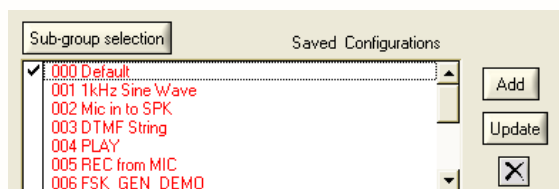


Figure 3-13 Default Path Settings

3.6 Memory Map

Memory Map generates the programming file to be burned into the ISD61S00 memory. The programming file is a binary file with .mem extension. On this panel users will see how each VP, VM, and message is assigned an address and/or index. To activate the process, click on 'Create Programming File' (A). The left hand side displays the memory map (B).

On the right hand side, the Memory Protection (C) decides how you want the memory contents protected. ISD61S00 provides three kinds of protection mechanisms:

- Read Protection:
 - Once enabled, digital-read command is not effective in those protected area.
- Write Protection:
 - Once enabled, digital-write and memory-erase commands are not effective in those protected area.
- Chip-Erase Protection:
 - Once enabled, chip-erase command is not effective.

If both Chip Erase Protect (CEP) and Write Protect (WP) are checked, it will lock the content from beginning of the memory to PMP, preventing ISD61S00 from any further modification.

The check boxes under 'Project Information' (D) allow users to add project information to the programming file. If users check the users comment box, whatever they type in the comment window will be appended to the end of the project information.

If the EV board is connected to PC, the 'Burn Device' button (E) will be effective. Once the programming file is created, users can click 'Burn Device' to download the programming file into the ISD61S00.

Please note that 'Date' (F) is part of the project, so clicking on the 'Create Programming File' (A) may change the project contents even though users don't change anything. When users open an existing project, the memory map will be loaded automatically.

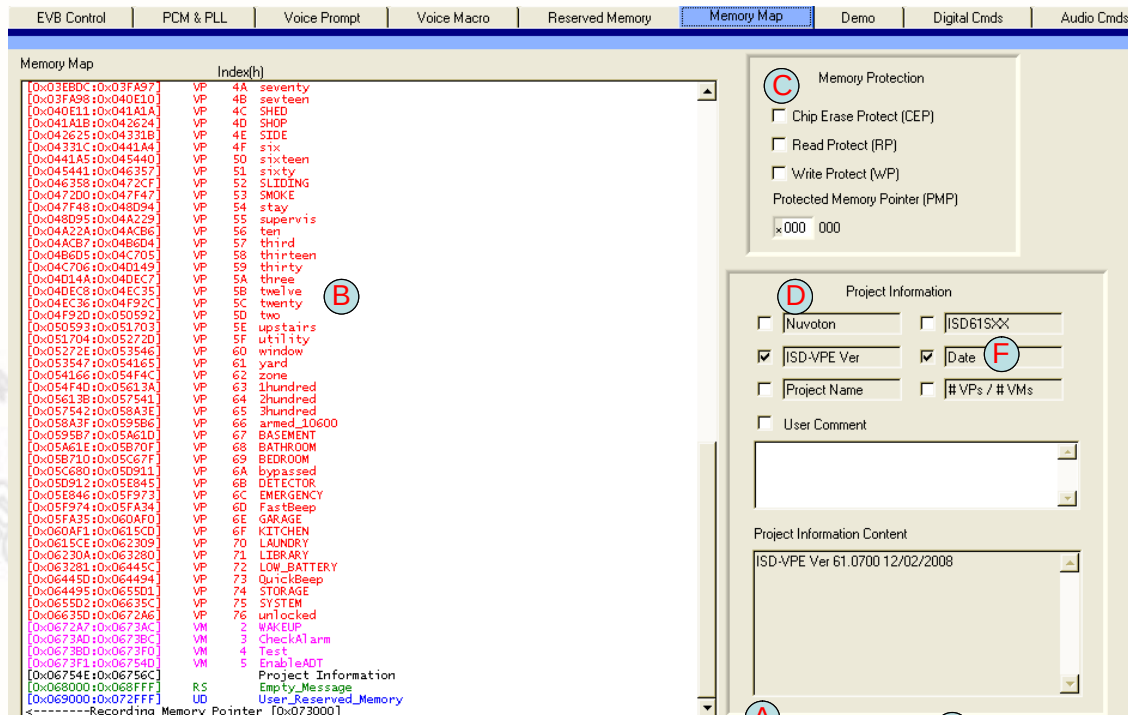


Figure 3-14 Memory Map

3.7 Archive and Clean

Under project menu there are two very useful functions.

1. Archive and,
2. Clean

These two functions are used to backup your project files and clean up unnecessary files. They are useful when you want to send an ISD61S00 project to other people.

The archiving function duplicates a project to an archive directory with exactly same project name and structure, and changes the project links to point to this directory so that all paths are stored relative to the project directory. For example, if you create an ISD61S00 project in a network share folder which is, say, mapped to T drive, you cannot just send this project to other people because “T” drive means nothing to other people. By archiving the project, that project is now self contained and can just be copied to another machine or to backup.

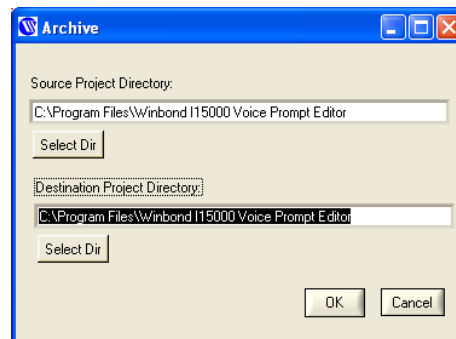


Figure 3-15 Archive

The Clean function deletes the contents in the BinFile and WavFile directories as they can be regenerated. The WavFile directory contains wave files of re-sampled and de-compressed voice prompts. The BinFile directory contains files of compressed voice prompts.

When a project is created, three sub-directories are created:

- BinFiles: where the compressed VPs are stored.
- WavFiles: where the re-sampled and de-compressed VPs are stored.
- WavSrc: where wave source of VPs are stored.

A project file has the prj extension. The binary programming file for the project has mem extension. A header file with txt extension is also included for C language firmware programming.

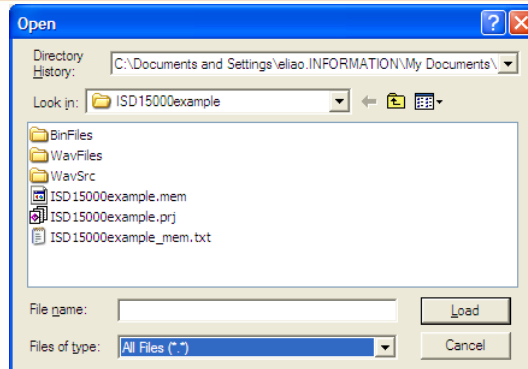


Figure 3-16 Open a Project

3.8 DTMF Detect

DTMF Detect can decode the digits in off-hook state and display them in (A).

To run the function, follow the steps below and refer to Figure 3-17:

- (1) Press the “GO” (B) button to load the default registers and the new registers will update to the panel immediately. Then the EVB will decode the default digits automatically and display them in (A).
- (2) Press the “Clear” (C) button to clear the decoded digits in (A) if needed; send more digits from other telephone through telephone line. And the decoded digits are also shown in (A).
- (3) If the decoded digits are not correct, tune the timing (D) (E) (F) according to Figure 3-18, tune the threshold (G), DTMF_TC (H) and Freq Deviation(I). And then Press “Write CFG” (J) button to change the conditions. Press “Read CFG” (K) to confirm the conditions if needed.
- (4) Send the digits again.
- (5) If the decoded digits are still not correct, repeat steps (3) & (4).

According to Figure 3-18, users can also monitor the immediate states: DTMF_RDY(L), DTMF_STATE(M) and DTMF_RX_DATA(N).

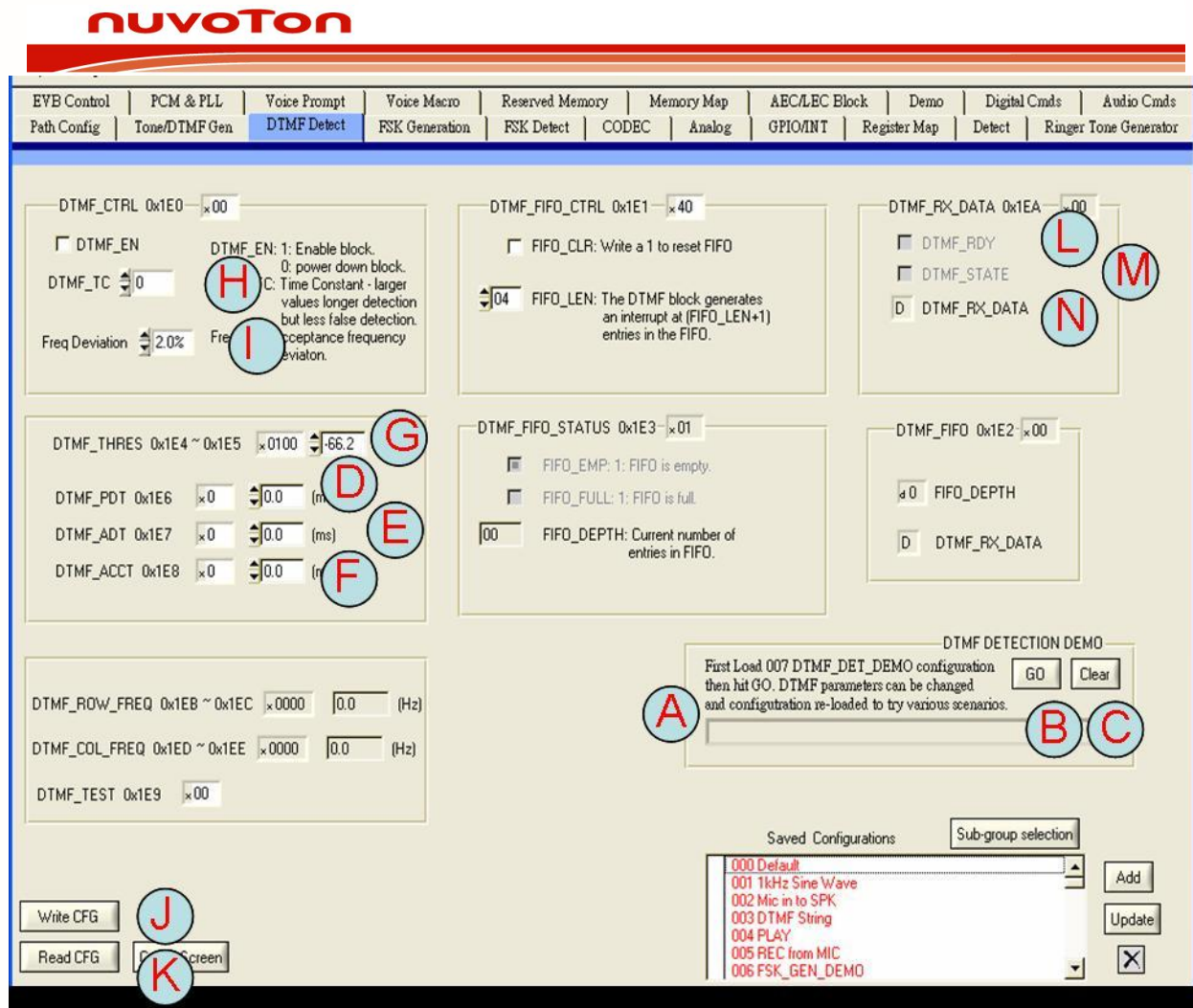


Figure 3-17 DTMF Detect screen

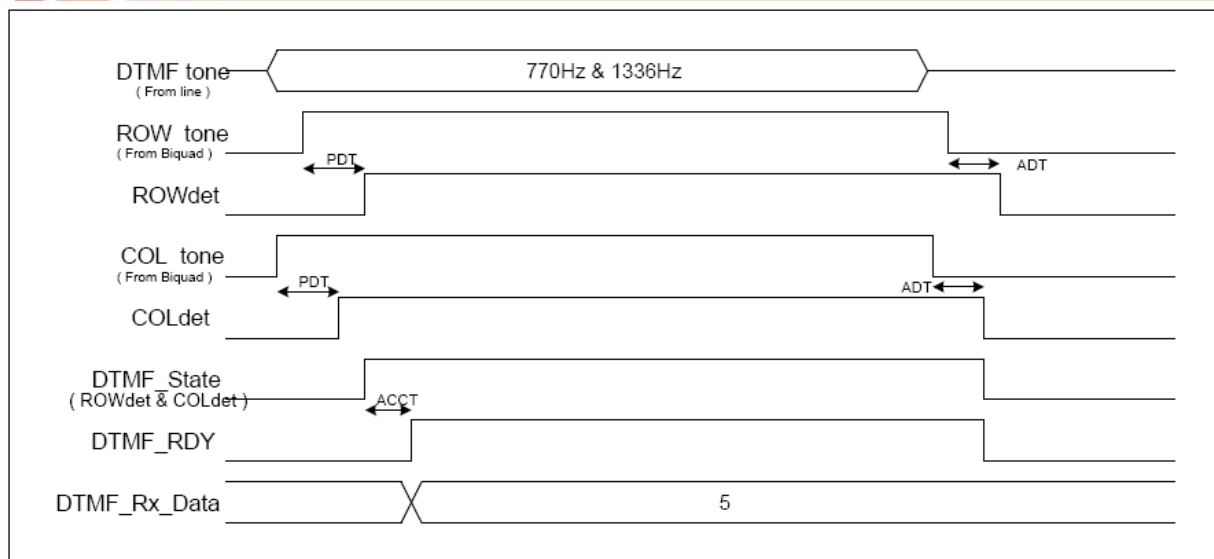


Figure 3-18 DTMF Detector Acquisition timing

3.9 Tone/DTMF Gen

Tone/DTMF generator operates in three different modes:

1. DTMF mode
2. Programmable mode
3. Continuous mode

If user set the DTMF_MODE to 1, the function is in the DTMF mode. If user set the DTMF_MODE to 0, the function is in the programmable and continuous mode. The setting of DTMF_MODE bit explained as following table:

Mode	TONE_FREQ_A TONE_FREQ_B	TONE_LEVEL_A TONE_LEVEL_B	TONE_ON_TIME TONE_OFF_TIME	TONE_LENGTH	TONES TO GEN
Continuous	User define	User define	Set: 0	Set: 0	Don't care
DTMF	Don't care	User define	User define	Number of digits in tone index registers.	Digits to send
Programmable	User define	User define	User define	Number of tones to generate	Don't care

For the DTMF mode,

- (1) Checking GEN_EN (A), DTMF_MODE (B) and CLK_EN (C)
- (2) Setting TONE_ON_TIME (D), TONE_OFF_TIME (E), TONE_LEV_A (H) and TONE_LEV_B (I).
- (3) To input the number of DTMF digits in the TONE_LENGTH (J) and the numeral you want to generate in TONES TO GEN (K).
- (4) Clicking Write CFG (L), the DTMF tone of the TONES TO GEN (K) would be generated.

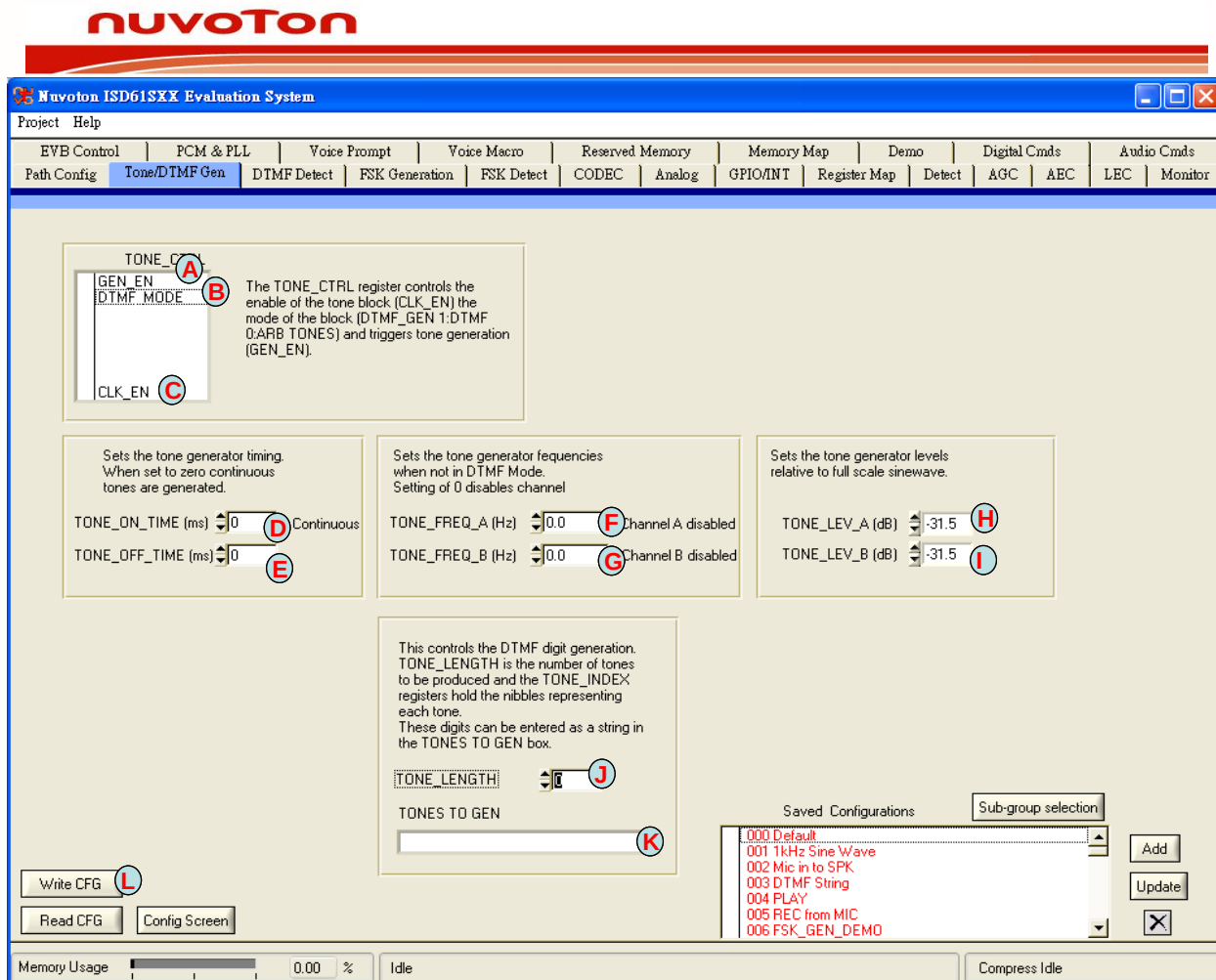


Figure 3-19 Tone/DTMF Gen

The frequency of DTMF is defaulted as following table:

		Column frequency			
Row Frequency		1209 Hz	1336 Hz	1477 Hz	1633 Hz
	697 Hz	0x01 _{hex}	0x02 _{hex}	0x03 _{hex}	0x0D _{hex}
		1	2	3	A
	770 Hz	0x04 _{hex}	0x05 _{hex}	0x06 _{hex}	0x0E _{hex}
		4	5	6	B
	852 Hz	0x07 _{hex}	0x08 _{hex}	0x09 _{hex}	0x0F _{hex}
		7	8	9	C
	941 Hz	0x0B _{hex}	0x0A _{hex}	0x0C _{hex}	0x00 _{hex}
		*	0	#	D

For Programmable mode,

(1) Checking GEN_EN (A) and CLK_EN (B)

- For continuous mode,

- The ISD61S00 has a FSK Detector supporting Bell202, Bell103, ITU-T V.21 and ITU-T V.23 standard FSK. The control panel is



(1) FSK Detect control registers:

- FSK detection parameter:

- FSKD_CDET(C), FSK_SYNC(D), FSKD_THRES(E), FSKD_TC(F), FSKD_ENERGY_HI(G), FSKD_ENERGY_LO(H), FSKD_MODE(K), FSKD_JUDGE(L).
- FSKD_FIFO_CTRL: control the FSK detection FIFO, to clear the FIFO, click 'CLEAR'(M) and Press 'Write CFG', the FSK detector FIFO will reset to a clean state.
- FSKD_Status: shows the FSK detect CDB and current demodulated data bit.
- FSKD_FIFO_STATUS: shows current FSK detection FIFO status.
- FSKD_FIFO_DOUT: current FIFO head.

To receive type 1 FSK CID message, the procedure is:

- Step1: click FSK_EN(A) and FSK_INT_MD(B) to checked state.
 Step2: configure FSKD_CDET(C), this value at least greater than 3.
 Step3: configure FSKD_SYNC(D), this vale at least greater than 2.
 Step4: configure FSKD_THRES(E), control the FSK detector sensitivity. The smaller, the more sensitive.
 Step5: configure FSKD_TC(F), FSKD_ENERGY_HI(G) and FSKD_ENERGY_LO(H). These 3 parameters only used when Bell202 or V.23 is chose. For other FSK spec (Bell103 and V.21), these 3 parameters has no effect.
 Step6: configure FSK_PARITY_EN(I), PARITY_TYPE(J), SPEC_SEL(K).
 Step7: configure FSK_CMP_CNT(L)
 Step8: clear FIFO by click 'CLEAR'(M), set the FIFO_LEN(N) to decide INT occur depth.
 Step9: Press 'GO'.
 Step10: configure SPEC_SEL(K) again.
 Step11: Press 'Write CFG'
 Step12: When FSK detector receives the message, it will display the message in the text box.

(2) Caller ID Detection : Demo area for Bell202 CID detection, the procedure is:

- Step1: click Bell202 Caller ID(T) in the right down configuration area.
 Step2: Press 'GO' button in 2nd area.
 Step3: When ISD61S receive a session of FSK CID signal, the Name(U) and Number(V) will display.

(3) FSK DETECTION DEMO : Demo for FSK generator and receive with Detector, the procedure is:

- Step1: select FSK spec in the SPEC_SEL(K) text box and press 'Write CFG'.
 Step2: prepare a text file with message in it.
 Step3: Press 'GO'(S) in 3rd area. The GUI will ask user to provide a file and then control ISD61S FSK generator to send this text in FSK spec selected in SPEC_SEL(K). When FSK detector receives the message, it will display the message in the text box.

(4) Dedicate area for security product. The procedure is:

- Step1: Select ACK Tone timing(R).
 Step2: Key in the dial number in the Dial Number text box(Q).
 Step3: Press 'GO'(P) and GUI will control ISD61S to communicate with a main machine.

3.11 FSK Generation

The FSK Generation supporting Bell202, Bell103, ITU-T V.21 and ITU-T V.23 standard FSK. The control panel is

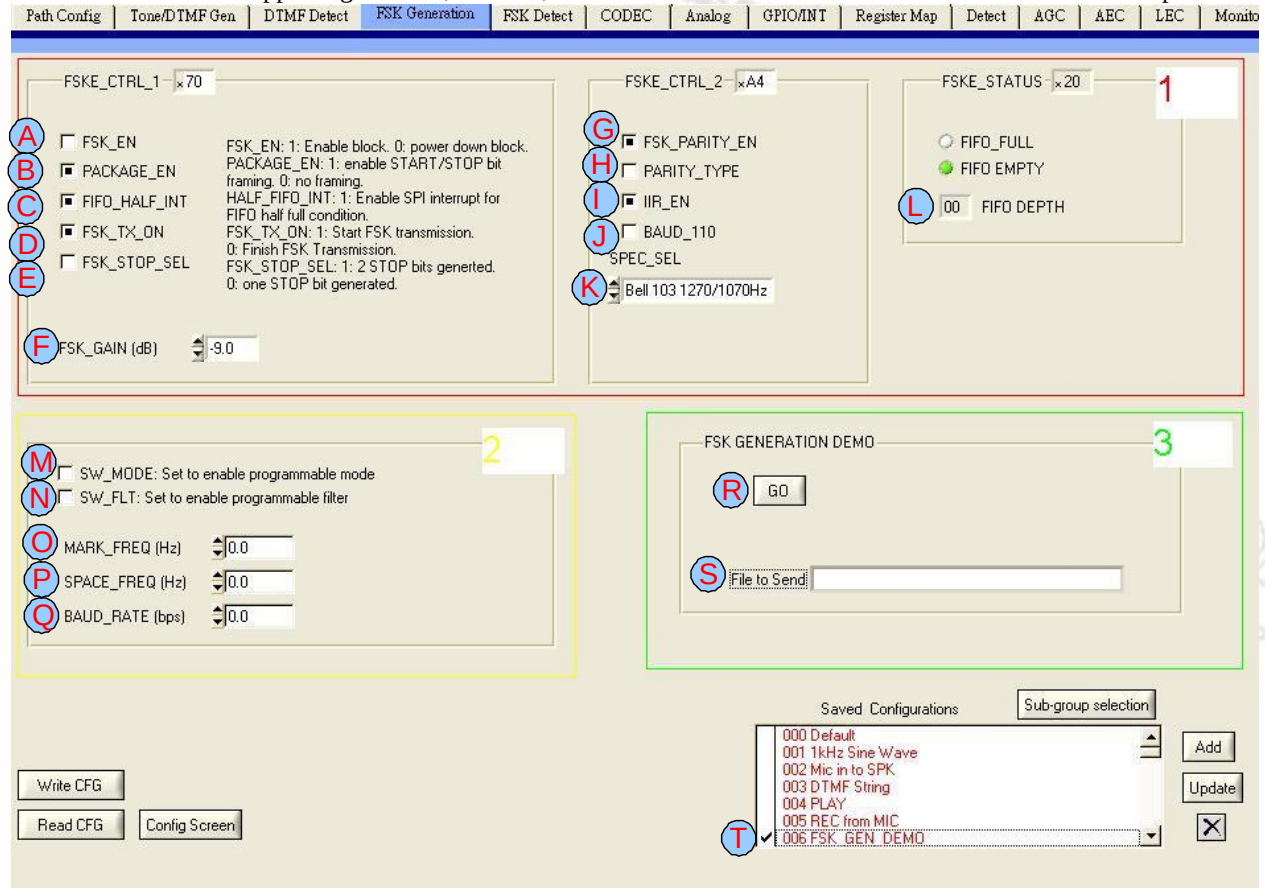
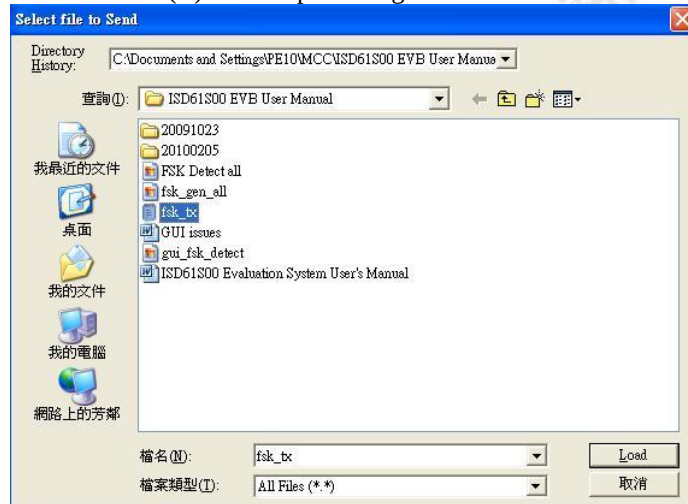


Figure 3-21 FSK Generation

Area 1 is FSK Encode control setting and status. Area 2 is specific customized mark and space baud rate setting. When SW_MODE(M) is checked, the SPEC_SEL(K) has no effect. Area 3 is for demo the transmit text with area 1 and area 2 setting.

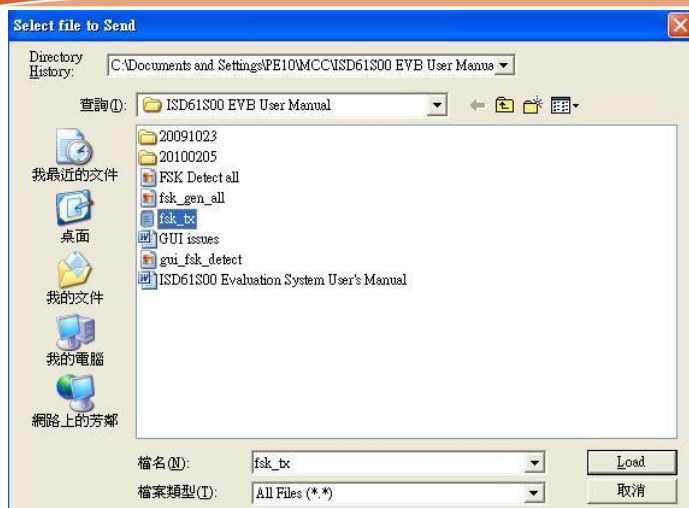
- To quickly select a FSK_GEN_DEMO:
Double click (T), if speaker is on, user should hear 1270Hz tone.
- To generate type 2 FSK signal:
 - Click FSK_EN(A)
 - Click PACKAGE_EN(B) to 1, this will insert start bit before data and stop bit behind data to construct a frame.
 - Click FIFO_HALF_INT(C), INT will occur when Transmit FIFO depth reaches half full condition.
 - Click FSK_TX_ON(D)
 - Configure FSK_STOP_SEL(E). 1 for two stop bits, 0 for 1 stop bit.
 - Configure FSK generation gain(F).
 - Configure FSK_PARITY_EN(G). 1 add parity bit. 0 no parity bit.
 - Configure FSK PARITY_TYPE(H). 0 for odd parity. 1 for even parity.
 - Click IIR_EN(I) to 1

10. Configure BAUD_110(J) to 1 if baud rate 110 is required.
11. Configure SPEC_SEL(K) to target FSK standard.
12. Prepare a text file with transmit content.
13. Press GO(R) a File input dialog will show



14. Select the file and Press 'Load'. The line interface will send FSK signal based on the text file.
15. To repeat transmit, press GO(R) again.

- To use SW Mode to generate type 2 FSK signal:
 1. Click FSK_EN(A)
 2. Click PACKAGE_EN(B) to 1, this will insert start bit before data and stop bit behind data to construct a frame.
 3. Click FIFO_HALF_INT(C), INT will occur when Transmit FIFO depth reaches half full condition.
 4. Click FSK_TX_ON(D)
 5. Configure FSK_STOP_SEL(E). 1 for two stop bits, 0 for 1 stop bit.
 6. Configure FSK generation gain(F).
 7. Configure FSK_PARITY_EN(G). 1 add parity bit. 0 no parity bit.
 8. Configure FSK PARITY_TYPE(H). 0 for odd parity. 1 for even parity.
 9. Click IIR_EN(I) to 1
 10. Click SW_MODE(M)
 11. Configure MARK_FREQ(O)
 12. Configure SPACE_FREQ(P).
 13. Configure BAUDRATE(Q).
 14. Prepare a text file with transmit content.
 15. Press GO(R) a File input dialog will show



16. Select the file and Press 'Load'. The line interface will send FSK signal based on the text file.

17. To repeat transmission, press GO(R) again.

3.12 Register Map

The Register Map shows the register status. There are two ways to modify the register values:

- (1) Modify many register values at the same time: To modify the value in the register map (A) at different addresses, click Write CFG (B), the modified value would be written into these registers. Clicking Read CFG (C), the register map shows the value you just modified. The color of modified value would be changed. The illustration is in the textbox (D).
- (2) Modify one register value at a time: to choose one address (E) and modify the value (F), button (V) would show the address and register name. Clicking Write Register (G), the register map (A) shows the changed value. Clicking Read Register (H) to check if the value is written into register.

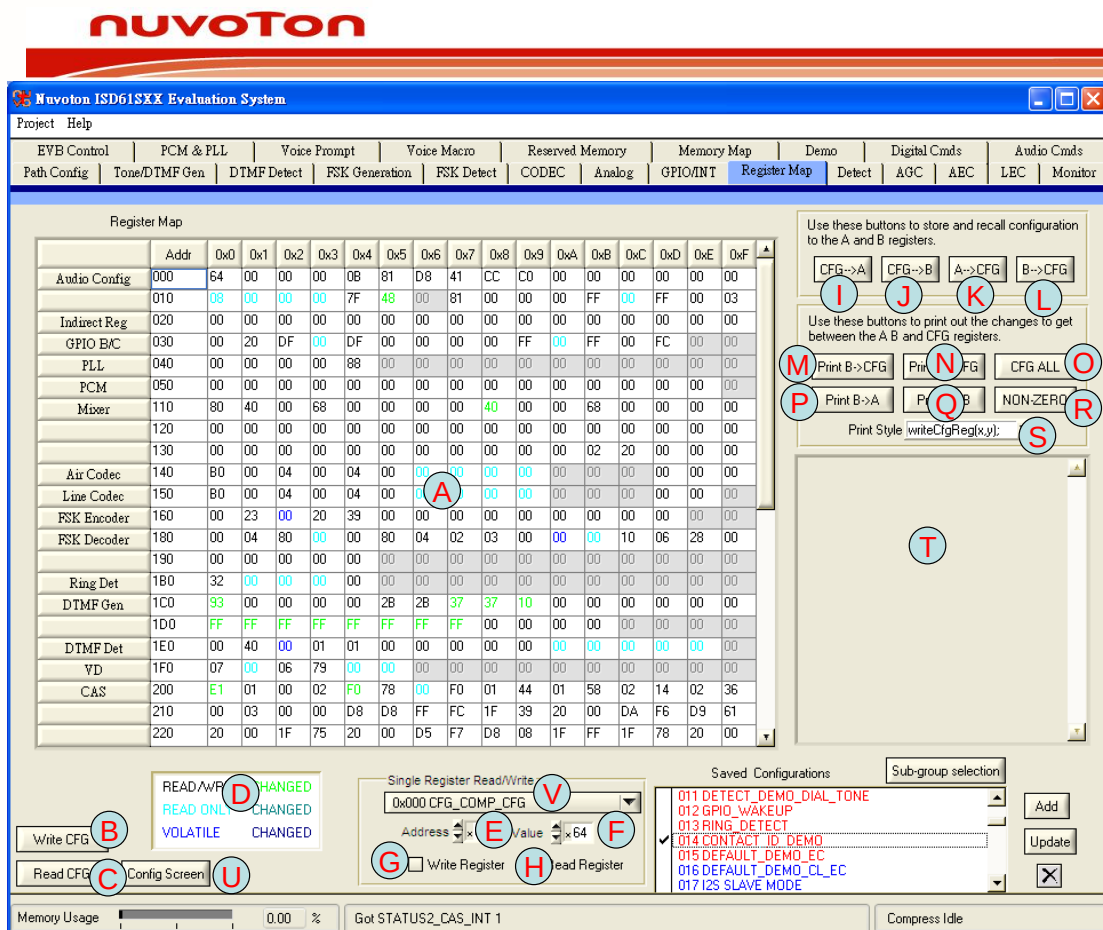


Figure 3-22 Register Map

There are two register storages in the panel. Clicking the CFG→A (I) button, the register values can be saved to register A. Clicking the CFG→B (J) button, the register values can be saved to register B. If you want to recover register A status, click A→CFG (K) button. If you want to recover register B status, click B→CFG (L) button.

Clicking the Print B→CFG (M) button, the text area (T) shows the different register values from register B to current configurations. Clicking the Print A→CFG (N) button, the text area (T) shows the different register values from register A to current configurations. The CFG ALL (O) button shows the register values in the text area (T) except the read only registers. The Print B→A (P) button shows the changed register values from register B to register A in the text area (T). The Print A→B (Q) button shows the changed register values from register A to register B. The NON-ZERO (R) button shows the non-zero register values. The Print Style (S) button can choose the style of address and register value showing in the text area (T). Clicking the Config Screen (U) button, GUI would jump to Path Config screen.

3.13 Analog

The Analog section contains the setting of MIC, SPK, PSTN line input and output, and AUX out.

- The microphone interface can operate on 3 modes: 1) Microphone Voltage gain mode; 2) Microphone Current gain mode; 3) Auxiliary input mode. The reference voltage and internal resistor can be configured by GUI.

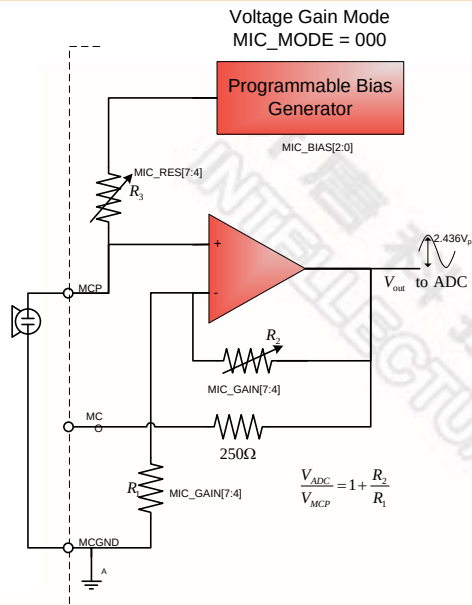


Figure 3-23 Voltage Gain Mode

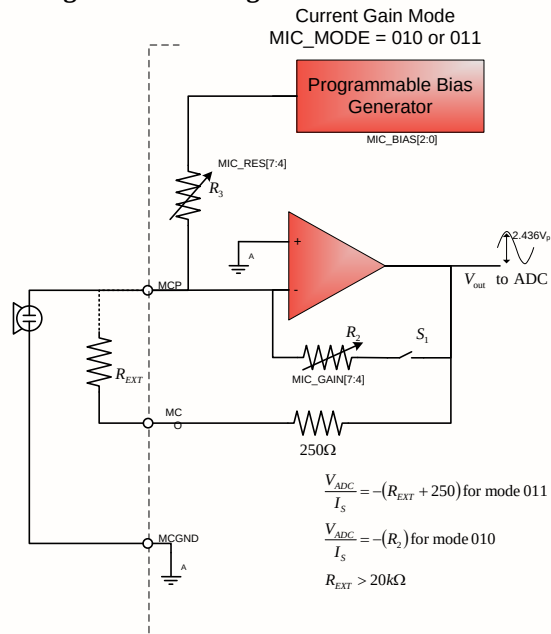


Figure 3-24 Current Gain Mode

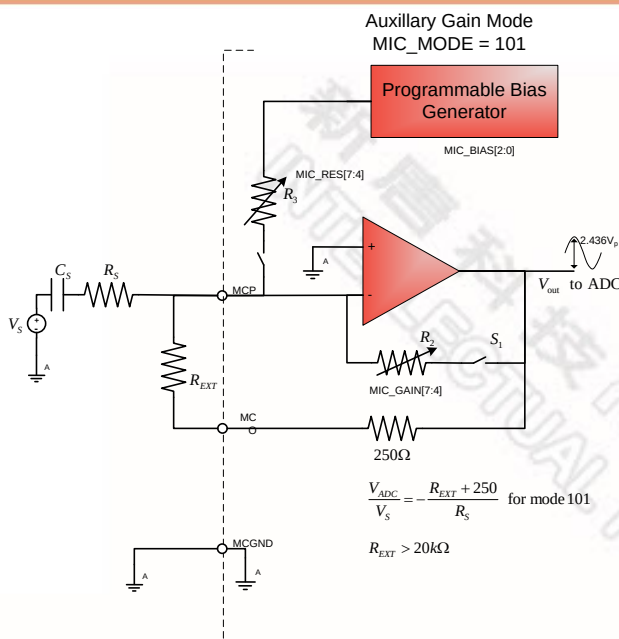


Figure 3-25 Auxiliary Gain Mode

- The PSTN line input contains two stage amplifier with independently settable gains and multiplexed input as described in figure 3-26.

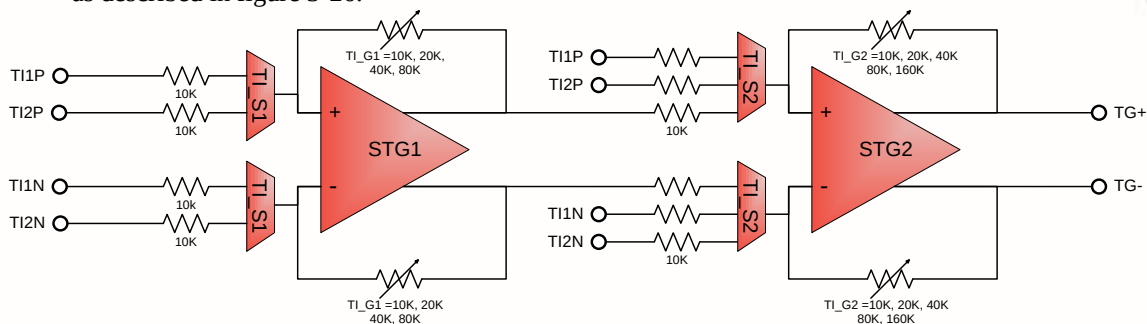


Figure 3-26 Two stage amplifier

- The analog output consists of speaker driver capable of driving 8 ohm and a differential line driver for PSTN interface capable of driving 120 ohm. These buffers are connected to the two DAC paths incorporating digital gain control. The ANA_OUT register controls the analog function of these two drivers allowing power control via SPK_EN and PO_EN and gain setting (including MUTE) via PO_GAIN and SPK_GAIN.

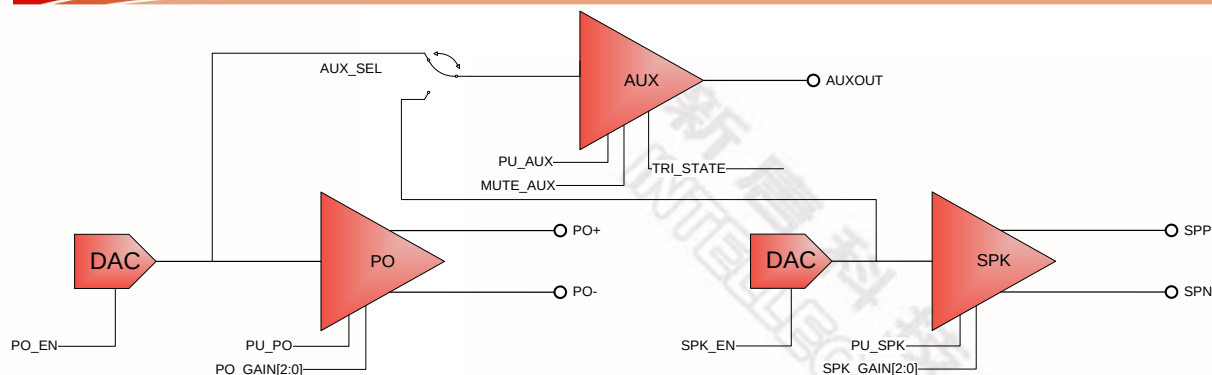


Figure 3-27 Analog output

The control panel of Analog section is:

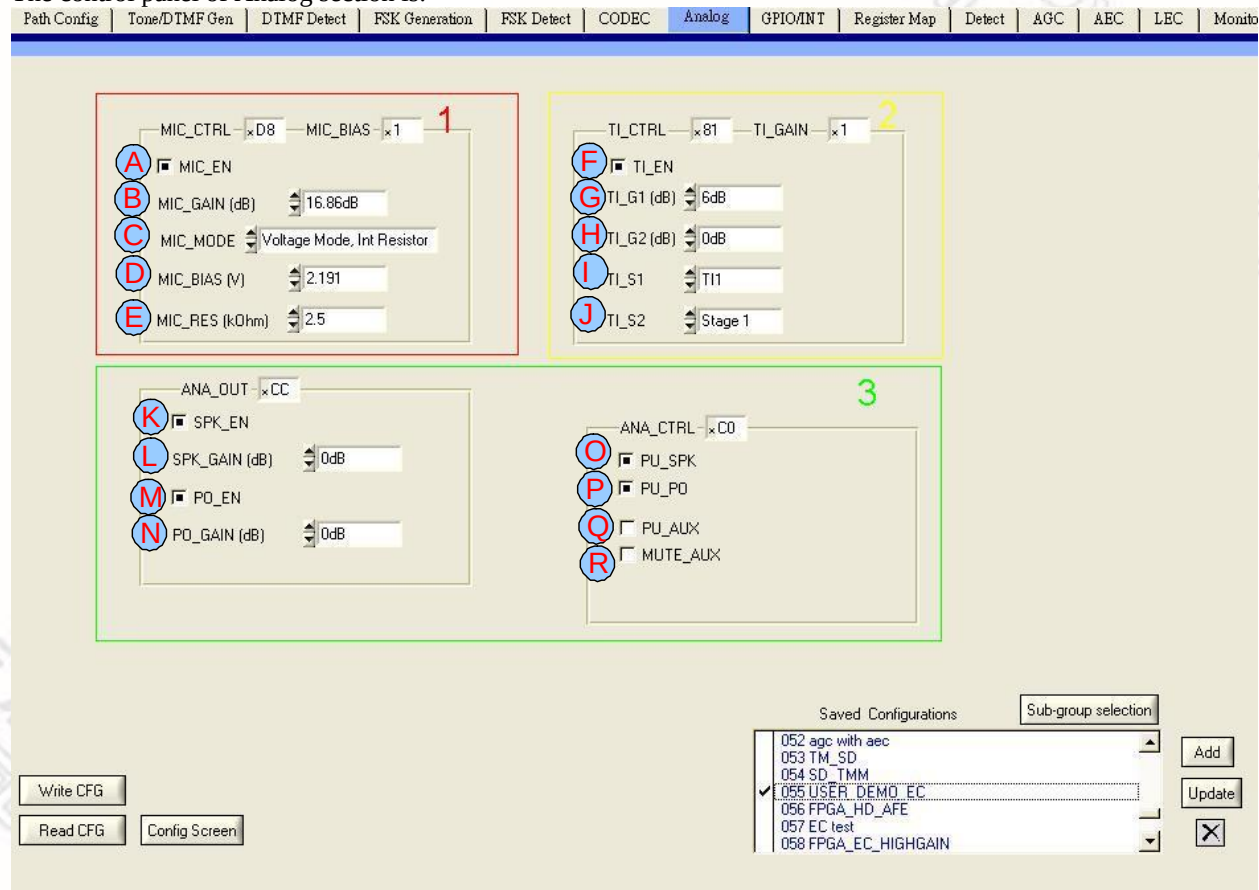


Figure 3-28 Analog setting

Figure 3-28 shows the analog input/output setting. Area 1 is for Microphone interface configuration. Area2 is for PSTN line input configuration. Area 3 is for PSTN line output and Speaker and AUX out.

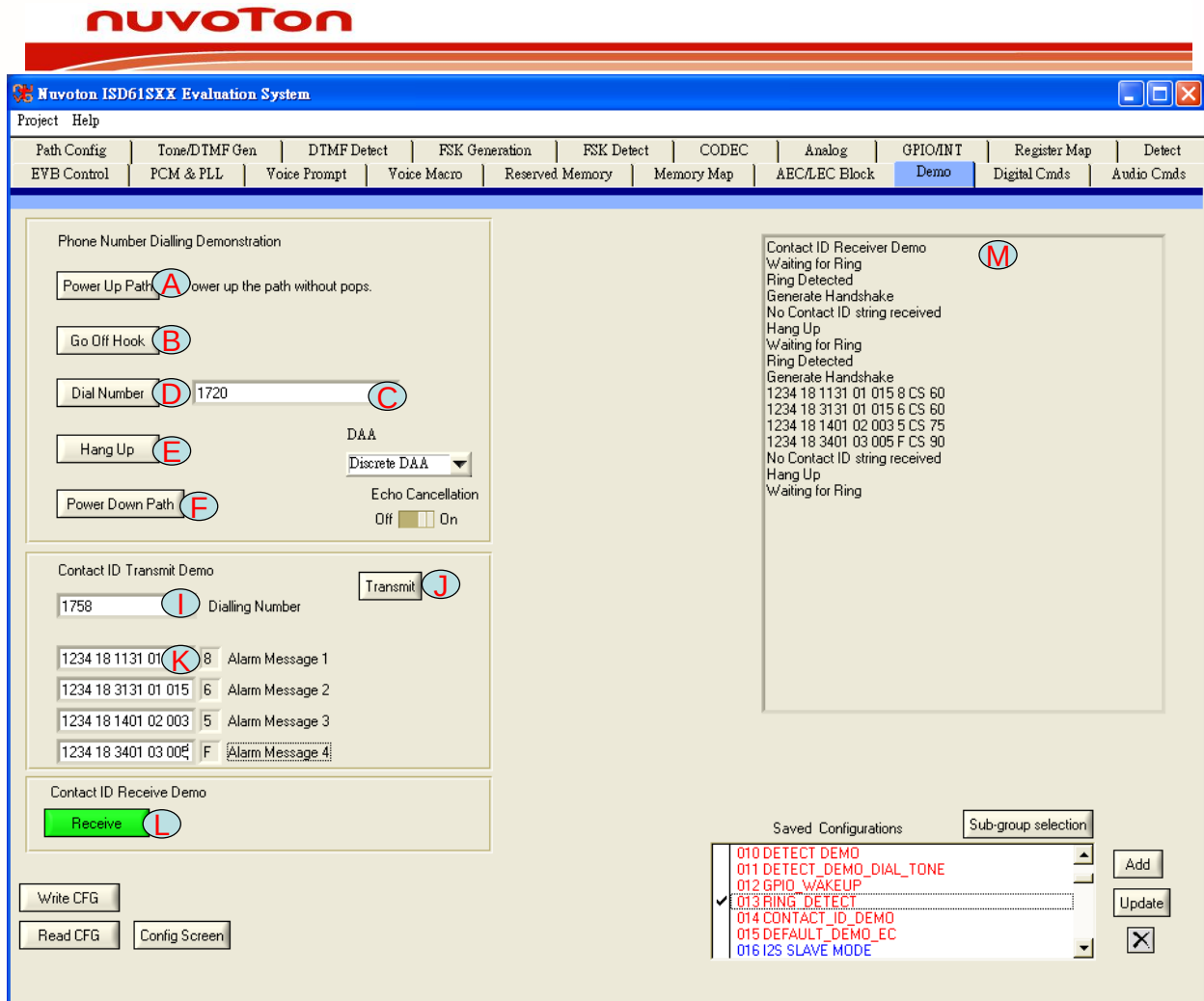
- To configure the MIC interface
 1. Click MIC_EN(A) to 1
 2. Configure MIC Gain(B)
 3. Configure MIC Mode(C)

4. Configure MIC_BIAS(D)
 5. Configure MIC_RES(E)
 6. Press 'Write CFG'
- To configure PSTN line input
 1. Click TI_EN(F) to 1
 2. Configure T1_G1(G), the gain of first stage amplifier
 3. Configure T1_G2(H), the gain of second stage amplifier
 4. Configure T1_S1(I), choose the input source of first stage amplifier
 5. Configure T2_S2(J), choose the input source of second stage amplifier
 6. Press 'Write CFG'
 - To configure Speaker output
 1. Click SPK_EN(K) to 1
 2. Configure speaker gain(L)
 3. Click PU_SPK(O)
 4. Press 'Write CFG'
 - To configure PSTN line output
 1. Click PO_EN(M) to 1
 2. Configure PO gain(N)
 3. Click PU_PO(P)
 4. Press 'Write CFG'
 - To configure AUX output
 1. Click PO_EN(M) to 1
 2. Configure PO gain(N)
 3. Click PU_AUX(Q) to 1
 4. Press 'Write CFG'

3.14 Demo

The following procedure demonstrates phone numbers dialing:

- (1) Clicking the Power Up Path (A) button: the path would be power up.
 - (2) Clicking Go off Hook (B) button: the 61S board would go to off hook status. If you don't click Power Up Path (A) before Go off Hook (B) button, the Go off Hook button would execute Power Up Path automatically.
 - (3) To input the dial numbers in the text area (C)
 - (4) Clicking the Dial Number (D) button. The 61S board would make a phone call according the number you input.
- After finishing the communication:
- (5) Clicking Hang Up (E) button.
 - (6) The Power Down Path (F) button is power down the device.



The Contact ID Transmit and Receive Demo:

This is the major security panel reporting method as specified by the SIA (security industry association) spec.

- For Contact ID Transmit Demo:

(1). Input the dial number in the text area (I)

(2). Clicking Transmit (J) button. The device would be waiting for the response of the other side.

If the other side sends a handshake response, the alarm messages would be transmitted. After transmitting alarm messages, the device would be hang up. The statuses are described in the text area (M).

- For Contact ID Receive Demo:

Clicking Receive (L) button, the device goes into the waiting for ring status.

If the device detects ringing, it will generate handshake signal and start to receive the messages from transmitter.

The messages are showing in the text area (M).

3.15 AEC/LEC

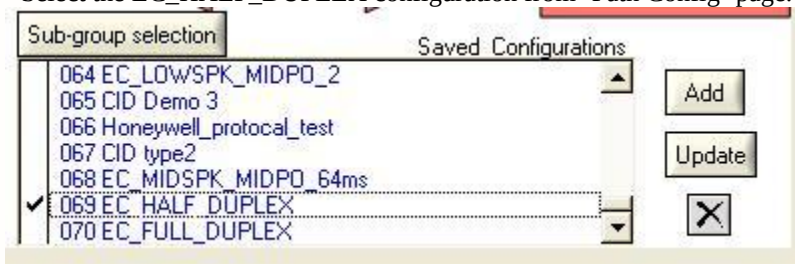
The Acoustic processing block performs a variety of functions to implement the full or half duplex echo cancellation. The ISD61SXX incorporates an Acoustic Echo Cancellation (AEC) block for the microphone and speaker CODEC and a Line Echo Cancellation (LEC) block for the PSTN interface. These blocks are very similar, with the main differences being that the AEC includes an AGC function and has a greater echo cancellation length.

The functions performed by the acoustic processing blocks include:

- **Echo Cancellation:** This is an adaptive estimation of the characteristics of the echo path to cancel echo introduced by speaker/microphone acoustic coupling and echo return paths from the environment.
- **Double Talk (DT) Detection:** When both line and acoustic paths are active it is important to detect this double talk conditions to halt adaptation of the acoustic echo.
- **Voice Detection (VT):** Estimates when the voice path is active.
- **AGC:** Automatic gain control to provide an optimal level from the microphone input.
- **Soft Clipping:** limits the input signal path to prevent hard saturation of the output.
- **Acoustic Suppression:** Used in half duplex mode to switch from one path to the other.
- **Noise Suppression:** Detect noise and suppress.

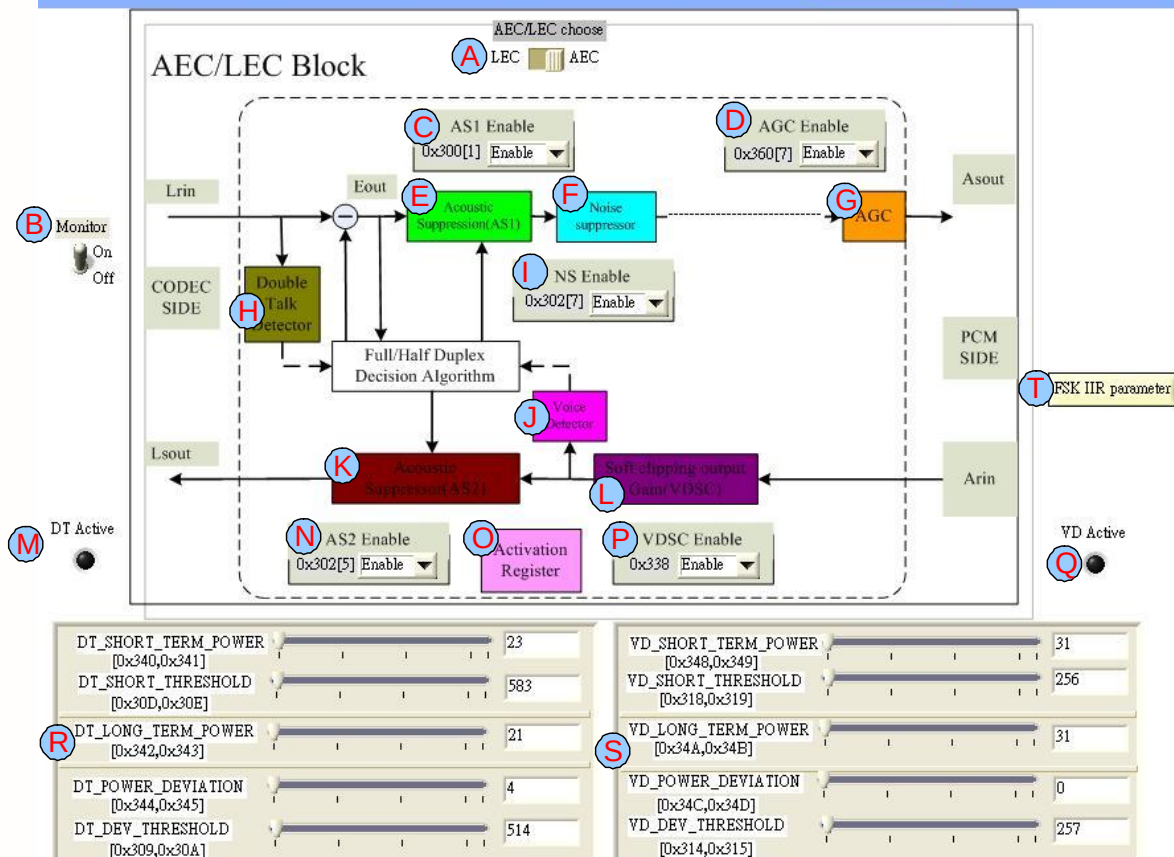
Full acoustic echo cancellation unit removes the echo signal inserted by the speaker coupling and space reflections. **Half duplex AEC** smoothly transitions the acoustic suppressors from one direction to the other based on the power estimations of the voice and double talk detectors.

- To use half duplex AEC:
Select the EC_HALF_DUPLEX configuration from 'Path Config' page.

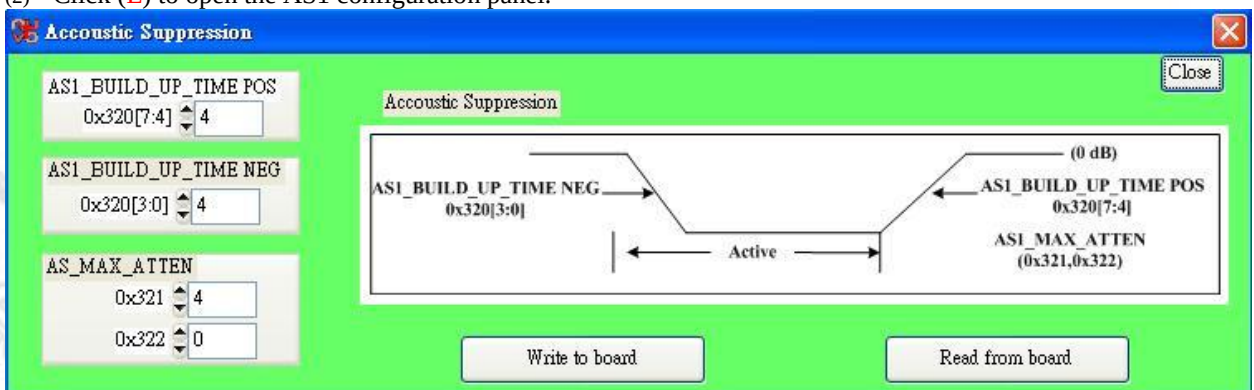


This will load the default values for the AEC/LEC half duplex mode setting.

- To tune the performance of AEC block
Switch to AEC/LEC block page and make sure the switch (A) is selected to AEC position.

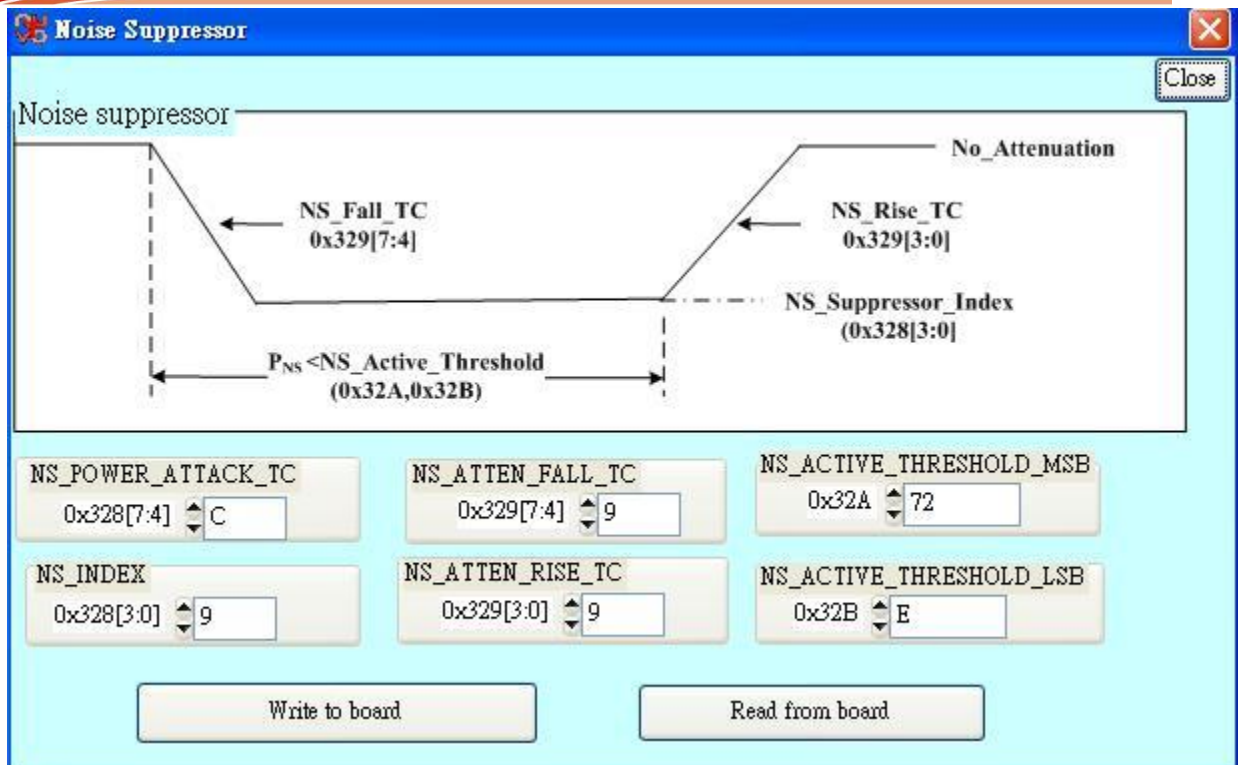


- To tune the Acoustic Suppressor 1 (AS1)
 - Click (C) to enable AS1.
 - Click (E) to open the AS1 configuration panel.



- Configure the parameters in the panel and Press 'Write to board'.

- To tune the Noise Suppressor (NS)
 - Click (I) to enable NS.
 - Click (F) to open the NS configuration panel.



(3) Configure the parameters in the panel and Press 'Write to board'.

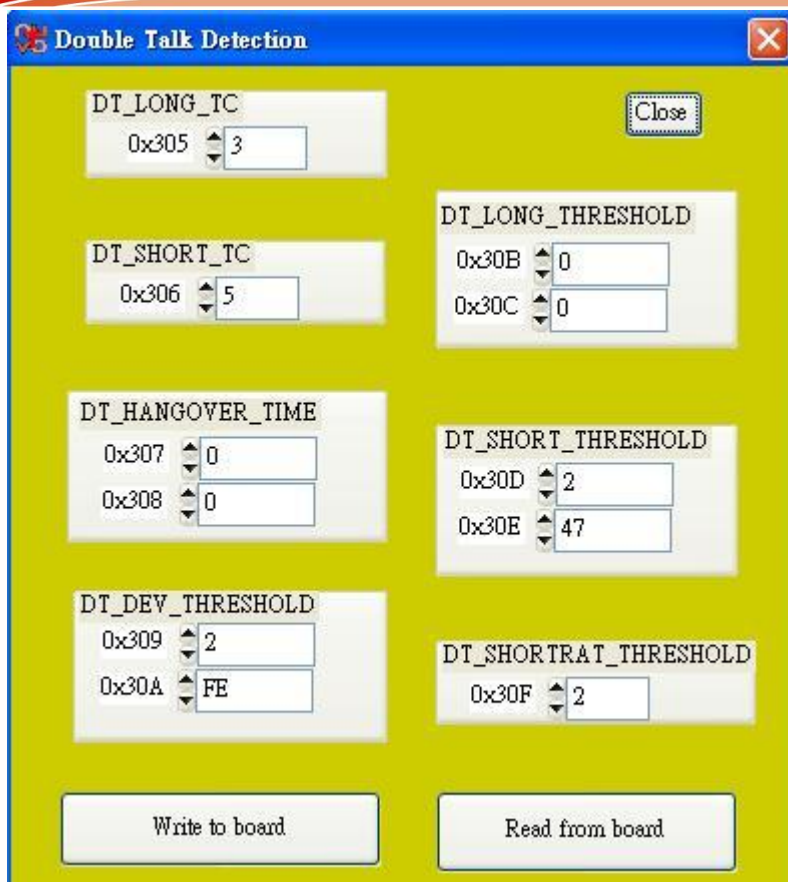
- To tune the Auto Gain Control (AGC)
 - (1) Click (D) to enable AGC.
 - (2) Click (G) to open the AGC configuration panel.

AGC [Close]

AGC_CTRL AGC_ONLY 0x360[6] Disable GAIN_LOW 0x360[5:3] 7.0 dB GAIN_HIGH 0x360[2:0] 7.0 dB	AGC_INC_DEC AGC_INC 0x363[6:4] 2 AGC_DEC 0x363[2:0] 1	AGC_STATE VOICE_ACTIVE 0x366[7] 0 HOLD_END 0x366[6] 1 MAX 0x366[5] 1 F_DEC 0x366[4] 0 INC 0x366[3] 0 DCY 0x366[2] 0 DEC 0x366[1] 1 ATK 0x366[0] 0
AGC_INIT_GAIN IG 0x361[7] 1 INITIAL_GAIN 0x361[6:0] 8	AGC_ATK_DCY AGC_DCY 0x364[6:4] 1 AGC_ATK 0x364[2:0] 1	AGC_ST_PWR_TC TC_H 0x367[7:4] 0 TC_L 0x367[3:0] 6
AGC_HOLD HOLD_TIME 0x362[7:5] 0 HOLD_CNT 0x362[4:0] 0	AGC_GAIN_READ 0x365 10	AGC_PEAK_CTRL XC_en 0x368[7] 0 (Disable) AGC_PEAK_REL 0x368[1:0] 1
AGC_PEAK AGC_PEAK(0x369,0x36A) x0003 -104.8 dBFS		
AGC_TARGETS AGC_TARG_CLIP(0x370,0x371) x4026 -30.1 dBFS AGC_TARG_HI(0x372,0x373) x0732 -49.1 dBFS AGC_TARG_LO(0x374,0x375) x0147 -64.1 dBFS AGC_TARG_RB(0x376,0x377) x0018 -86.8 dBFS	AGC_NOISE AGC_NOISE_THD_BIAS 0x378 0 AGC_NOISE_THD_HI(0x379,0x37A) x3A44 -30.9 dBFS AGC_NOISE_THD_LO(0x37B,0x37C) x293E -33.9 dBFS AGC_NOISE_RB(0x37D~0x37F) x000DA7 -43.5 dBFS	
Write to board		Read from board

(3) Configure the parameters in the panel and Press 'Write to board'.

- To tune the Double Talk Detector (DT)
- (1) Click (H) to open the DT configuration panel.



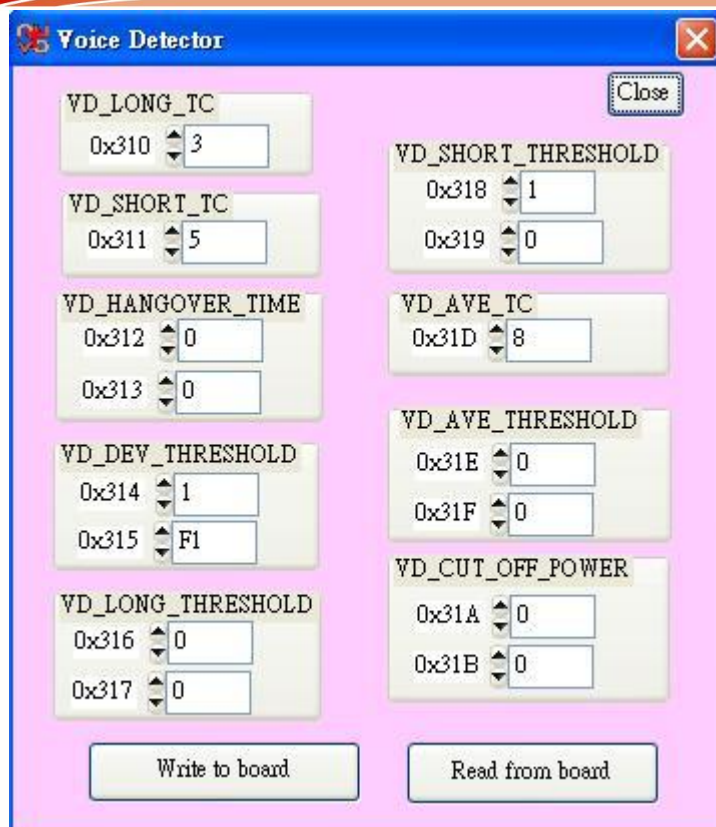
The image shows a software window titled "Double Talk Detection" with a yellow background and a blue title bar. It contains several configuration sections, each with a label, a memory address, and a value field with up/down arrows. A "Close" button is in the top right. At the bottom are "Write to board" and "Read from board" buttons.

Parameter	Address	Value
DT_LONG_TC	0x305	3
DT_SHORT_TC	0x306	5
DT_HANGOVER_TIME	0x307	0
	0x308	0
DT_DEV_THRESHOLD	0x309	2
	0x30A	FE
DT_LONG_THRESHOLD	0x30B	0
	0x30C	0
DT_SHORT_THRESHOLD	0x30D	2
	0x30E	47
DT_SHORTTRAT_THRESHOLD	0x30F	2

(2) Configure the parameters in the panel and Press 'Write to board'.

- To tune the Voice Detector (VD)

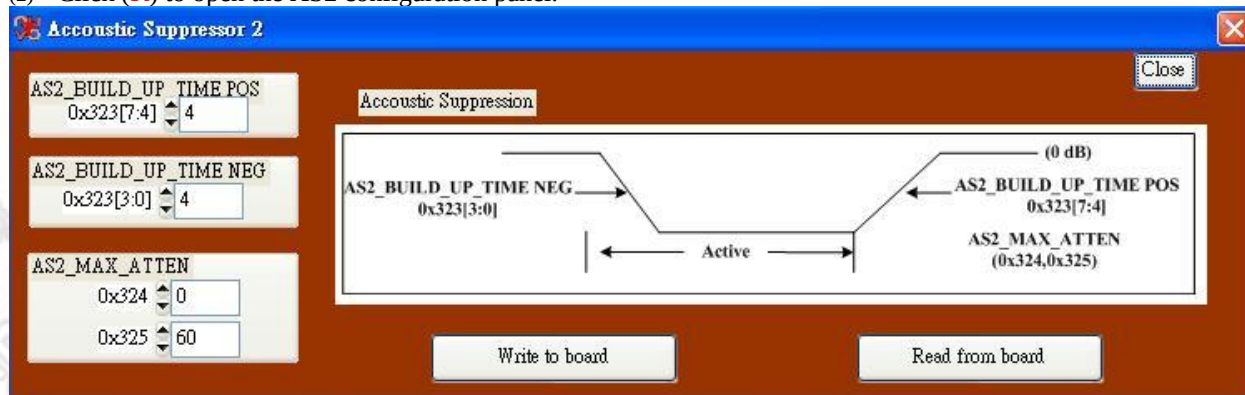
(1) Click (J) to open the VD configuration panel.



(2) Configure the parameters in the panel and Press 'Write to board'.

- To tune the Acoustic Suppressor 2 (AS2)

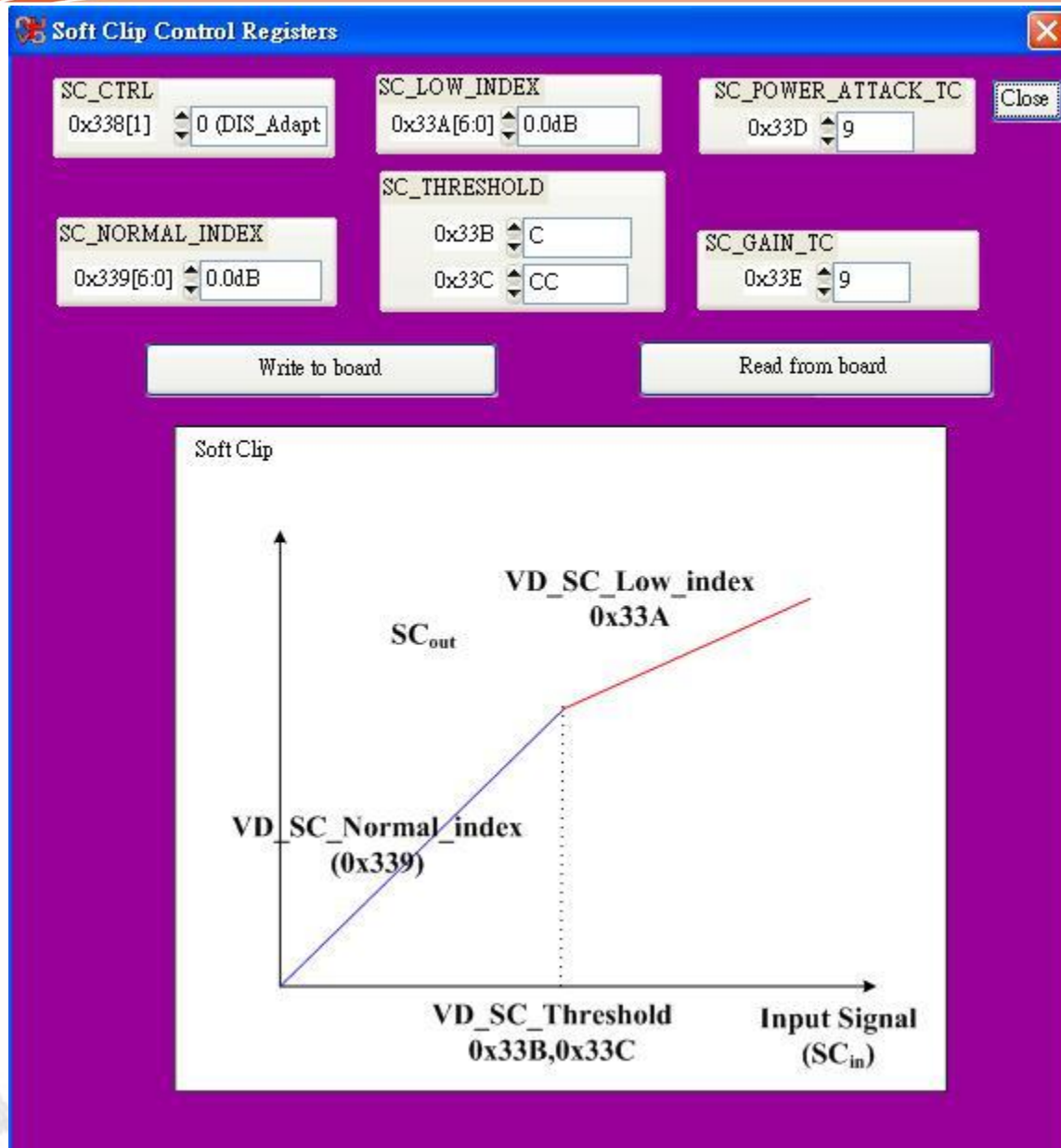
- (1) Click (N) to enable AS2.
- (2) Click (K) to open the AS2 configuration panel.



(3) Configure the parameters in the panel and Press 'Write to board'.

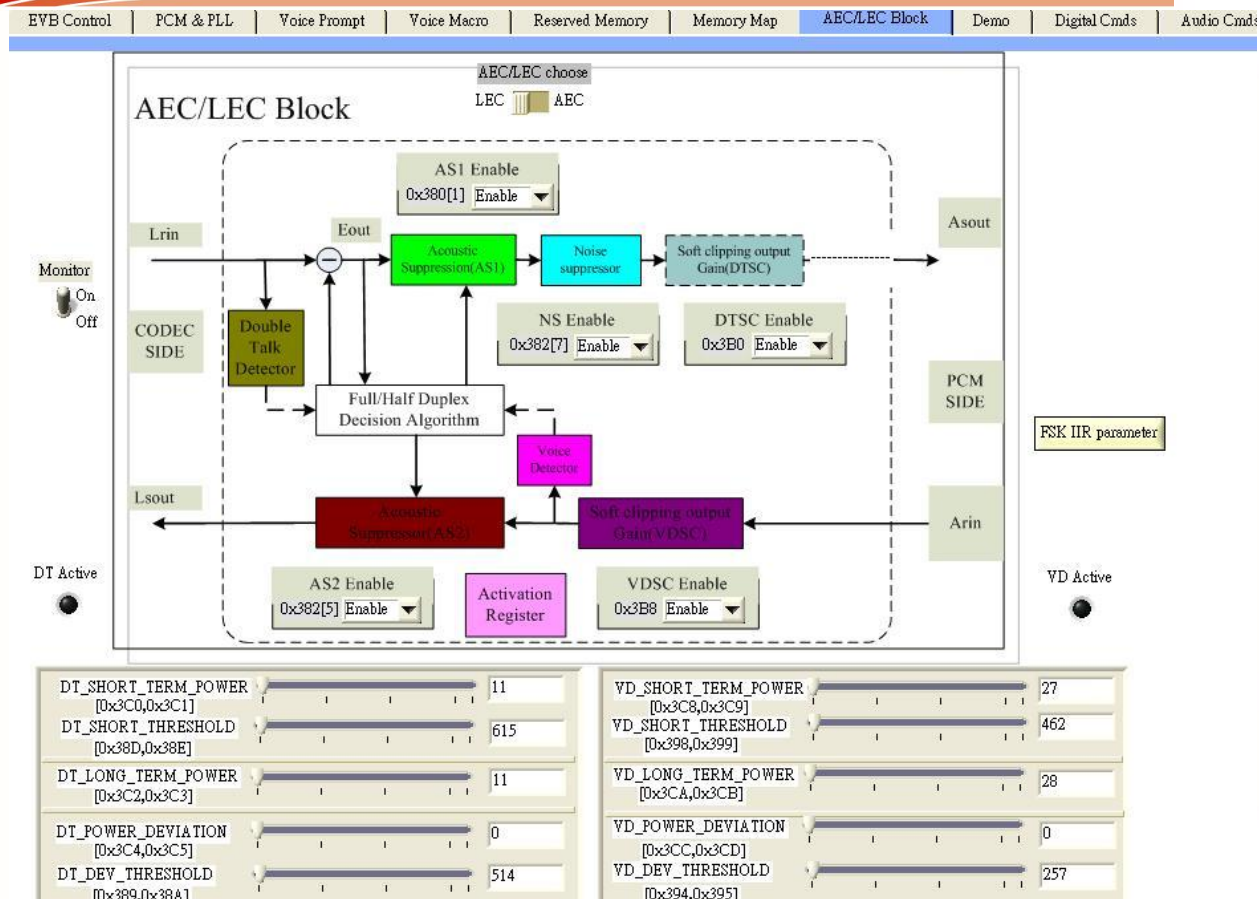
- To tune the Soft Clipping(SC)

- (1) Click (P) to enable SC.
- (2) Click (L) to open the SC configuration panel.



(3) Configure the parameters in the panel and Press 'Write to board'.

- To tune the performance of LEC block
Switch to AEC/LEC block page and make sure the LEC switch is selected.



The function block operations are similar to previous AEC procedure.

- To use full duplex AEC/LEC:
Select the EC_FULL_DUPLEX configuration from 'Path Config' page.



This will load the default values for the AEC/LEC full duplex mode setting.
The AEC/LEC detail tuning is similar to previous procedure.

3.16 AGC

The AGC function is in the AEC/LEC block (A). To set the AGC Enable, and double click the AGC icon, the AGC setting shows on the screen.

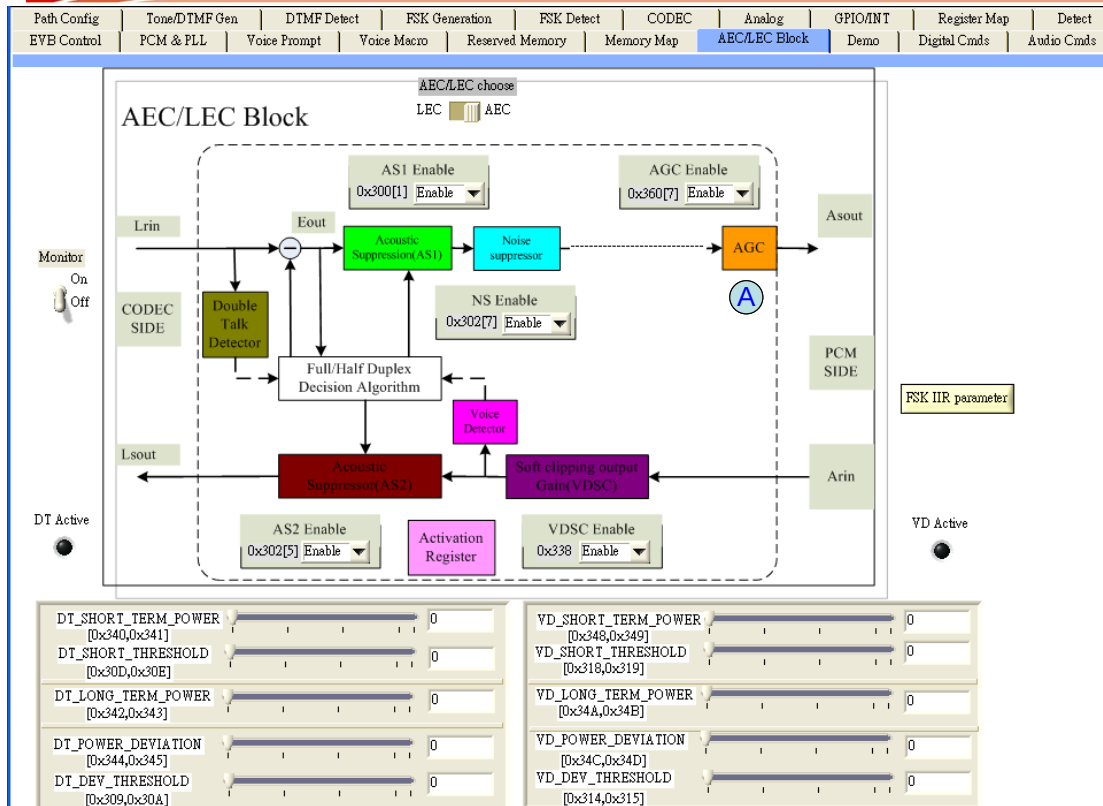


Figure 3-29 AEC/LEC screen

The AGC screen displays the following parameters and their current values:

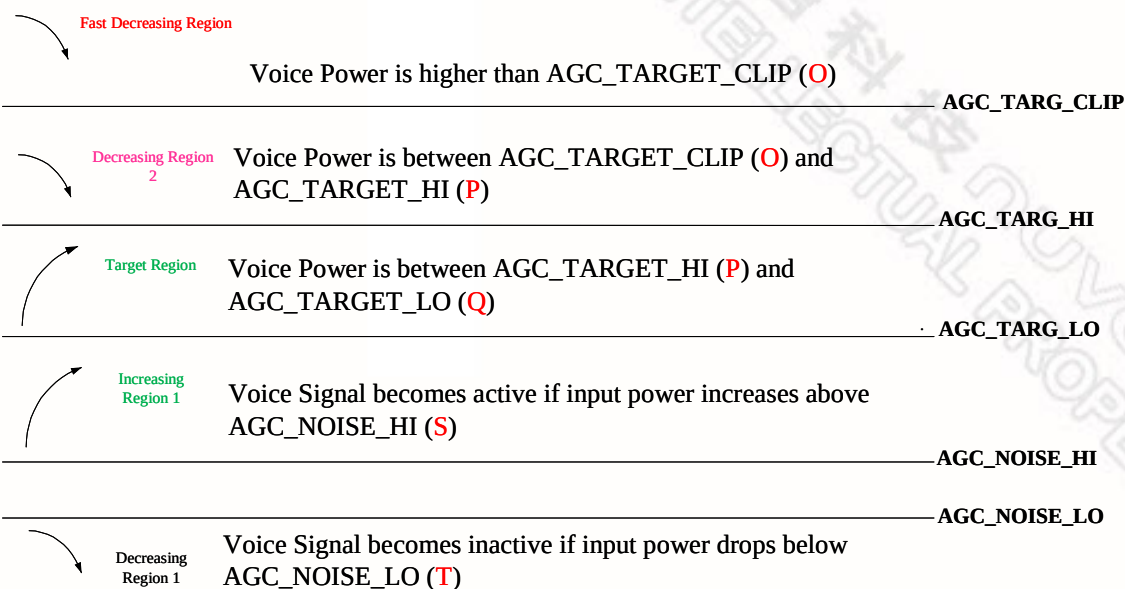
- AGC_CTRL**: AGC_ONLY 0x360[6] **Enable** (A), GAIN_LOW 0x360[5:3] **3.9 dB** (B), GAIN_HIGH 0x360[2:0] **17.6 dB** (C).
- AGC_INIT_GAIN**: IG 0x361[7] **0** (D), INITIAL_GAIN 0x361[6:0] **8** (E).
- AGC_HOLD** (F): Hold Time **0000 ms**, AGC_HOLD 0x362[7:5] **0**, AGC_HOLD_CNT 0x362[4:0] **0**.
- AGC_INC_DEC** (G): INC_Time **004 us**, AGC_INC 0x363[6:4] **0**, DEC_Time **004 us**, AGC_DEC 0x363[2:0] **0**.
- AGC_ATK_DCY** (H): DCY_Time **0016 us**, AGC_DCY 0x364[6:4] **0**, ATK_Time **0016 us**, AGC_ATK 0x364[2:0] **0**.
- AGC_GAIN_RB** (I): **0.0 dBFS**, AGC_GAIN_READ 0x365 **x10**.
- AGC_STATE** (J): VOICE_ACTIVE 0x366[7] **0**, HOLD_END 0x366[6] **1**, MAX 0x366[5] **1**, F_DEC 0x366[4] **0**, INC 0x366[3] **0**, DCY 0x366[2] **0**, DEC 0x366[1] **1**, ATK 0x366[0] **0**.
- AGC_ST_PWR_TC** (K): TC_H 0x367[7:4] **0**, TC_L 0x367[3:0] **0**.
- AGC_PEAK_CTRL**: XC_en 0x368[7] **0 (Disable)** (L), REL_Time **04 us** 0x368[1:0] **0** (M).
- AGC_PEAK** (N): AGC_PEAK(0x369,0x36A) **x222E** **-11.5 dBFS**.
- AGC_TARGETS**: AGC_TARG_CLIP(0x370,0x371) **x7777** **-0.6 dBFS** (O), AGC_TARG_HI(0x372,0x373) **x6644** **-1.9 dBFS** (P), AGC_TARG_LO(0x374,0x375) **x3333** **-8.0 dBFS** (Q), AGC_TARG_RB(0x376,0x377) **x2251** **-11.4 dBFS** (R).
- AGC_NOISE**: AGC_NOISE_THD_HI (0x379,0x37A) **x4444** **-40.1 dB** (S), THD_BIAS 0x378 **1**, AGC_NOISE_THD_LO (0x37B,0x37C) **x3333** **-40.6 dB** (T), AGC_NOISE_RB(0x37D~0x37F) **000000** **-96.0 dBFS** (U).

Buttons at the bottom: Write to board (V), Read from board (W).

Figure 3-30 AGC screen

1. If the AGC_ONLY (A) is enabled, the AEC block is bypassed and AGC operates from AEC input.
2. The GAIN_LOW (B) and GAIN_HIGH (C) are setting the min and max gain.
3. The INITIAL_GAIN (E) determines the initial gain if the IG (D) set to 1.
4. The AGC_HOLD (F) determines how long the AGC will hold gain constant after entering target region before increasing gain.
5. The AGC_INC and AGC_DEC of AGC_INC_DEC (G) determine the time constant at which the gain increase and decrease.
6. The AGC_ATK_DCY (H) controls the decay time constant (AGC_DCY) and attack time constant (AGC_ATK) when in the target region.
7. The AGC_GAIN_RB (I) shows the current AGC gain. The AGC_STATE (J) shows some states of AGC function.
8. The AGC_ST_PWR_TC (K) determines the time constant for calculating the short term noise energy. The short term noise energy responds faster to signal change when the time constant larger.

9. The REL_Time (M) is the release time of peak detector. If the XC_en (L) is enabled, the AGC_GAIN would be changed when the input signal is in the zero crossing.
10. The AGC_PEAK (N) reads back the peak detector value.
11. The AGC_TARGETS and AGC_NOISE set the high or low limit of target and noise region. When the signal short term energy increases above AGC_NOISE_THD_HI (S), the AGC is activated. The illustration is as below figure:



12. The AGC_TARG_RB (R) is read only for the current AGC output signal level.
13. The AGC_NOISE_RB (U) shows the current noise power estimated.
14. Clicking Write to board (V), the settings would be written into registers.
15. Clicking Read from board (W) can read the register settings and check if the settings written into registers or not.

3.17 GPIO/INT

GPIO and interrupt are the two functions of the panel and are described as below, respectively.

3.17.1 GPIO

The GPIO pins are divided into three port group:

- (1) PORTA consists of GPIO [7:0]
- (2) PORTB consists of GPIO [15:8]
- (3) PORTC consists of GPIO [16]

You can set the configurations in “GPIO configuration Bits” (A) or “GPIO configuration Registers” (B).

The GPIO pin can be either an input pin or an output pin by enabling or disabling “OE” (C).

1. The GPIO pin is an input pin: If the “OE” (C) is enabled, the corresponding GPIO pin is an output pin. And you can change the output state by setting “DO” (D) and the driving strength by setting “DS” (E).

2. On the other hand, if the “OE” (C) is disabled, the corresponding GPIO pin is an input pin. And you can change the pin state by enabling “PE” (F) and setting “PS” (G).
The port state can be checked by the value of “DI” (H).

If “AF” (I) is enabled, the corresponding GPIO pin may have an alternate function and are shown as below.

Port	GPIO Pin	Alternate Function	Description
PORTA<4>	GPIO<4>	SDO	I2S Serial Data Out
PORTA<5>	GPIO<5>	WS	I2S WS
PORTA<6>	GPIO<6>	SCK	I2S Serial Clock
PORTA<7>	GPIO<7>	SDI	I2S Serial Data In
PORTB<0>	GPIO<8>	SDO1	PCM1 Tx Data
PORTB<1>	GPIO<9>	SDI1	PCM1 Rx Data

3.17.2 Interrupt

There are sixteen enables and they are in “Interrupt Enable 0” (J) and “Interrupt Enable 1” (K). If you enable the interrupt, then the corresponding interrupt state will be shown in “Audio Cmds” panel.

If the interrupt state shows the detected signal, the GPIO should be an input pin and the interrupt is enabled. For example, the ringing interrupt detect the ringing signal from GPIO 14 in J1 or J11 on the EVB board, so the GPIO14 should be an input pin; “RINT_DET_IN” (L) should be GPIO14; RING_INT (M) should be enabled, then the ringing interrupt state will be shown in “Audio Cmds” panel.

The ISD61SXX can be waked up by a voltage level change. To do it, you should set the GPIO pin as an input pin; enable the interrupt (N) of the corresponding pin and enable the corresponding wakeup interrupt (O).

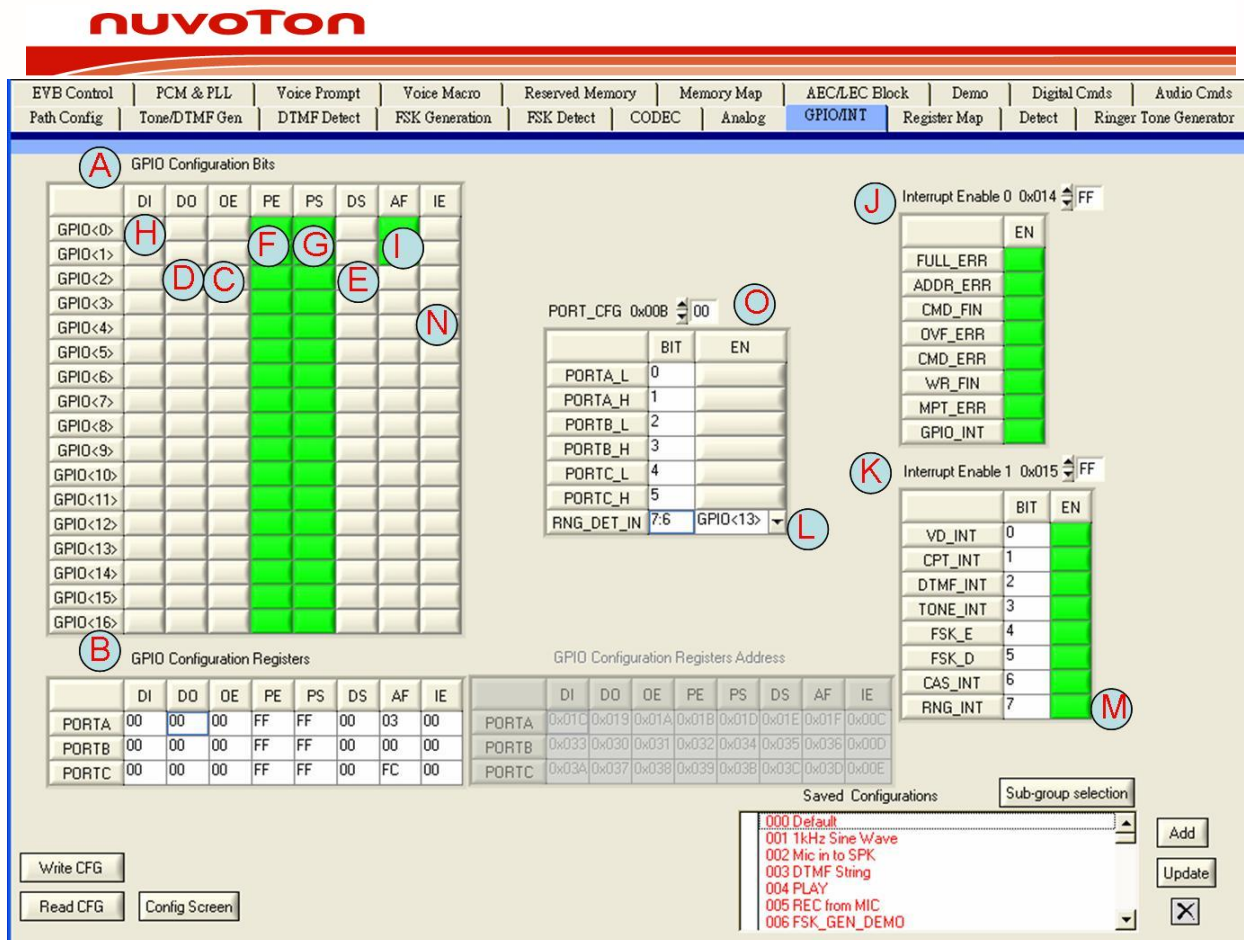


Figure 3-31 GPIO/INT screen

3.18 Detect

The detect panel has several function settings including CAS detect, ATD, RING detect, CPT detect and VD.

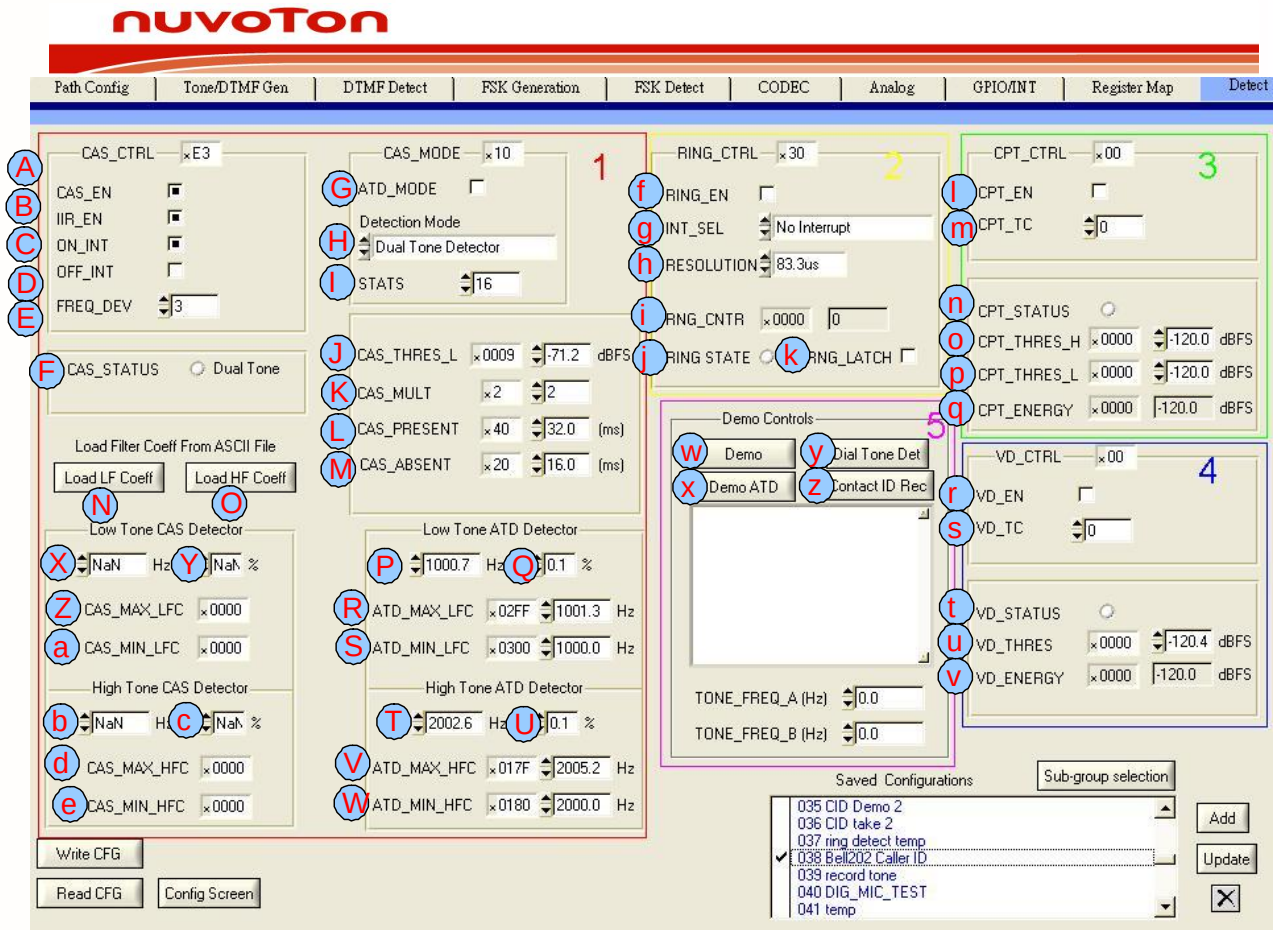


Figure 3-32: The Detect panel has 4 setting area and one demo area.

Area 1 is for CAS detect and ATD setting. The ATD_MODE(G) and Detection Mode(H) will select the Tone detect engine to operate as dual tone detect (ie CAS detect) or ATD (arbitrary tone detect). Area 2 is for ring detect. Area 3 is for CPT (Call Progress Tone) detect. Area 4 is for VD (Voice detect). Area 5 demo the function of tone detects with tone generation to generate the desired tones and feedback to tone detect.

3.18.1 CAS detect

To use CAS detect:

1. Click CAS_EN (A) to 1, enable the CAS module.
2. Click IIR_EN (B) to 1, enable the IIR filter.
3. Click ON_INT (C) to 1, this will issue interrupt when CAS detected.
4. Click OFF_INT (D) to zero. If OFF_INT is 1, another interrupt will issued when CAS is absent.
5. Configure FREQ_DEV (E) to 3, select 3% deviation.
6. Click ATD_Mode (G) to 0. For CAS mode, this check box must be 0.
7. Configure Detection Mode (H) to Dual Tone Detector.
8. Configure the STATS (I) to decide number of sine wave cycles to judge.
9. Configure CAS_THRES_L (J) to decide low threshold
10. Configure CAS_MULT (K) to decide high threshold multiplier to low threshold.
11. Configure CAS_PRESENT (L) to decide present time.
12. Configure CAS_ABSENT (M) to decide absent time.
13. Press 'Write CFG'

If CAS tone is detected, press 'Read CFG', the Dual Tone status (F) will change to green.

3.18.2 ATD

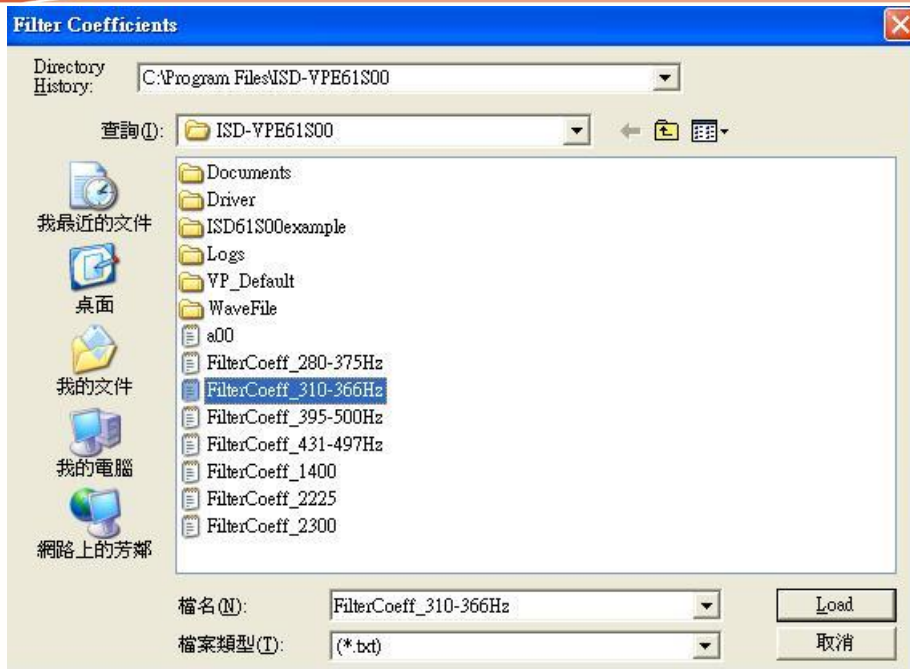
The screenshot displays the ATD configuration window with the following sections and settings:

- Path Config:** Tone/DTMF Gen, DTMF Detect, FSK Generation, FSK Detect, CODEC, Analog, GPIO/INT, Register Map, Detect.
- CAS_CTRL (x E1):** CAS_EN (A), IIR_EN (B), ON_INT (C), OFF_INT (D), FREQ_DEV (E).
- CAS_MODE (x E7):** ATD_MODE (G), Detection Mode (H) set to Low & High Tone Detector, STATS (I) set to 7.
- CAS_THRES_L (x 0100):** -42.1 dBFS.
- CAS_MULT (x 1):** 1.
- CAS_PRESENT (x 20):** 16.0 (ms).
- CAS_ABSENT (x 10):** 8.0 (ms).
- CAS_STATUS (F):** Low Tone (green), High Tone (green).
- Load Filter Coeff From ASCII File:** Load LF Coeff (N), Load HF Coeff (O).
- Low Tone CAS Detector:** 2129.8 Hz (X), 3.3 % (Y).
- CAS_MAX_LFC (x 015D):** 015D.
- CAS_MIN_LFC (x 0175):** 0175.
- High Tone CAS Detector:** 2750.9 Hz (b), 3.4 % (c).
- CAS_MAX_HFC (x 010E):** 010E.
- CAS_MIN_HFC (x 0121):** 0121.
- Low Tone ATD Detector:** 350.2 Hz (P), 3.2 % (Q).
- ATD_MAX_LFC (x 03A2):** 361.3 Hz.
- ATD_MIN_LFC (x 03DF):** 339.1 Hz.
- High Tone ATD Detector:** 440.2 Hz (T), 3.1 % (U).
- ATD_MAX_HFC (x 02E4):** 454.1 Hz.
- ATD_MIN_HFC (x 0314):** 426.4 Hz.
- RING_CTRL (x 30):** RING_EN, INT_SEL (No Interrupt), RESOLUTION (83.3us), RING_CNTR (x 0000), RING_STATE, RING_LATCH.
- CPT_CTRL (x 87):** CPT_EN, CPT_TC (7).
- CPT_STATUS:** CPT_THRES_H (x 03A3) -55.0 dBFS, CPT_THRES_L (x 0126) -65.0 dBFS, CPT_ENERGY (x 1A99) -37.7 dBFS.
- VD_CTRL (x 87):** VD_EN, VD_TC (7).
- VD_STATUS:** VD_THRES (x 03A4) -55.0 dBFS, VD_ENERGY (x 0066) -74.2 dBFS.
- Demo Controls:** Demo, Dial Tone Det, Demo ATD, Contact ID Rec.
- TONE_FREQ_A (Hz):** 0.0.
- TONE_FREQ_B (Hz):** 0.0.
- Saved Configurations:** 006 FSK_GEN_DEMO, 007 DTMF_DET_DEMO, 008 1kHz SINE to PO, 009 OFF HOOK EC, 010 DETECT DEMO, 011 DETECT DEMO DIAL TONE (checked), 012 GPIO_WAKEUP.
- Buttons:** Write CFG, Read CFG, Config Screen, Add, Update, X.

Figure 3-33: An example of ATD setting and detected result.

To use ATD:

- Phase1: load band pass filter coefficients
 1. Click CAS_EN (A) to 1, enable the CAS module.
 2. Click IIR_EN (B) to 0, disable the IIR filter.
 3. Click ATD_Mode (G) to 1. For ATD mode, this check box must be 1.
 4. Configure Detection Mode (H) to Low&High Tone Detector.
 5. Click Load LF Coeff (N), a file input dialog will ask user to select the coefficient file.



Choose the desired filter coefficient file and press 'Load'.

6. Click Load HF Coeff (O), another file input dialog will ask user to select the coefficient file.



Choose the desired filter coefficient file and press 'Load'.

7. Press 'Write CFG' and the Band pass Filter coefficients will write in the register.

- Phase2: configure the detect parameter
 1. Click IIR_EN (B) to 1, enable the IIR filter.

- Click ON_INT (C) to 1, this will issue interrupt when tone detected.
- Click OFF_INT (D) to zero. If OFF_INT is 1, another interrupt will issued when tone is absent.
- Configure the STATS (I) to decide number of sine wave cycles to judge.
- Configure CAS_THRES_L (J) to decide low threshold
- Configure CAS_MULT (K) to 1
- Configure CAS_PRESENT (L) to decide present time.
- Configure CAS_ABSENT (M) to decide absent time.
- Configure desired low tone frequency (P) and deviation percentage (Q). The ATD_MAX_LFC (R) and ATD_MIN_LFC (S) will be calculated by GUI automatically.
- Configure desired high tone frequency (T) and deviation percentage (U). The ATD_MAX_HFC (V) and ATD_MIN_HFC (W) will be calculated by GUI automatically.
- Press 'Write CFG'

If low or high tone is detected, press 'Read CFG', the low tone or high tone status (F) will change to green.

3.18.3 Ring Detect

Figure 3-34: An example of Ring Detect setting.

To use Ring Detect:

- Click RING_EN (f) to 1.
- Configure INT_SEL (g) to choose INT occur edge.
- Configure the RESOLUTION (h) to decide the counter increasing step.
- Press 'Write CFG'

If ring is applied in the Tip/Ring, the RING_INT will occur.

3.18.4 CPT Detect

The screenshot displays the 'Detect' configuration window of the ISD61S00 software. The 'CPT_CTRL' (x87) panel is highlighted with a green border and contains the following settings:

- CPT_EN (I):** Enabled (checkbox checked).
- CPT_TC (m):** 7.
- CPT_STATUS (n):** Green indicator.
- CPT_THRES_H (o):** 03A3, -55.0 dBFS.
- CPT_THRES_L (p):** 0126, -65.0 dBFS.
- CPT_ENERGY (q):** 0DF7, -43.3 dBFS.

Other visible settings include:

- CAS_CTRL (x A1):** CAS_EN, IIR_EN, ON_INT, OFF_INT, FREQ_DEV (1).
- CAS_MODE (x F0):** ATD_MODE, Detection Mode (Low & High Tone Detector), STATS (16).
- RING_CTRL (x 30):** RING_EN, INT_SEL (No Interrupt), RESOLUTION (83.3us), RNG_CNTR (0000), RING_STATE, RNG_LATCH.
- VD_CTRL (x 87):** VD_EN, VD_TC (7), VD_STATUS, VD_THRES (03A4, -55.0 dBFS), VD_ENERGY (004B, -76.9 dBFS).
- Demo Controls:** Demo, Dial Tone Det, Demo ATD, Contact ID Rec, Test ARB Detector 350Hz 440Hz, TONE_FREQ_A (Hz) (0.0), TONE_FREQ_B (Hz) (0.0).
- Saved Configurations:** List of configurations including '010 DETECT DEMO' (selected).

Figure 3-35: An example of CPT Detect setting.

To use CPT:

1. Click CPT_EN (I) to 1
2. Configure CPT_TC (m). Larger values imply a large time constant, greater stability but less response to change in energy level.
3. Configure CPT_THRES_H (o).
4. Configure CPT_THRES_L (p).
5. Press 'Write CFG'.

If signal energy (q) is over the CPT_THRES_H (o), Press 'Read CFG', the CPT_STATUS (n) will change to green.

3.18.5 VD

Path Config | Tone/DTMF Gen | DTMF Detect | FSK Generation | FSK Detect | CODEC | Analog | GPIO/INT | Register Map | Detect

CAS_CTRL ☒ A1
CAS_EN ☒
IIR_EN ☐
ON_INT ☒
OFF_INT ☐
FREQ_DEV

CAS_MODE ☒ F0
ATD_MODE ☒
Detection Mode
STATS

RING_CTRL ☒ 30
RING_EN ☐
INT_SEL
RESOLUTION
RNG_CNTR 0
RING_STATE ☐ RNG_LATCH ☐

CPT_CTRL ☒ 87
CPT_EN ☒
CPT_TC

CAS_STATUS ☐ Low Tone ☐ High Tone

Load Filter Coeff From ASCII File
Load LF Coeff Load HF Coeff

Low Tone CAS Detector
 Hz +/- %
CAS_MAX_LFC
CAS_MIN_LFC

High Tone CAS Detector
 Hz +/- %
CAS_MAX_HFC
CAS_MIN_HFC

Low Tone ATD Detector
 Hz +/- %
ATD_MAX_LFC Hz
ATD_MIN_LFC Hz

High Tone ATD Detector
 Hz +/- %
ATD_MAX_HFC Hz
ATD_MIN_HFC Hz

Write CFG Read CFG Config Screen

Demo Controls
Demo Dial Tone Det Demo ATD Contact ID Rec

TONE_FREQ_A (Hz)
TONE_FREQ_B (Hz)

CPT_STATUS ☐
CPT_THRES_H dBFS
CPT_THRES_L dBFS
CPT_ENERGY dBFS

VD_CTRL ☒ 87
VD_EN ☒
VD_TC

VD_STATUS ☒
VD_THRES dBFS
VD_ENERGY dBFS

Saved Configurations Sub-group selection
008 1kHz SINE to PD
009 OFF HOOK EC
010 DETECT DEMO
011 DETECT DEMO_DIAL_TONE
012 GPIO_WAKEUP
013 RING_DETECT
014 CONTACT_ID_DEMO
Add Update

Figure 3-36: An example of VD setting.

To use VD:

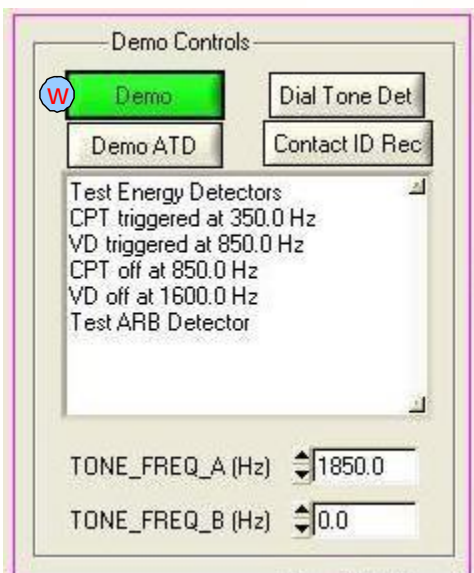
1. Click VD_EN (r) to 1
2. Configure VD_TC (s)
3. Configure VD_THRES (u)
4. Press 'Write CFG'

If signal energy (v) between 750~1400Hz is detected and over the VD_THRES (u), Press 'Read CFG', the VD_STATUS (t) will change to green.

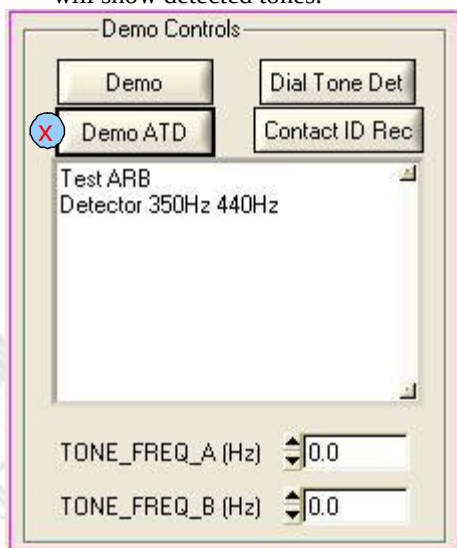
3.18.6 Demo

The Demo area shows some demos.

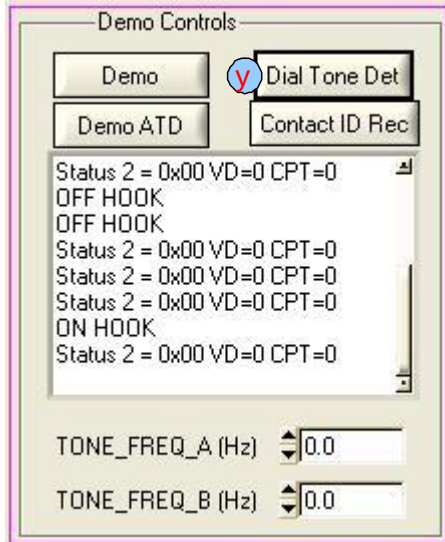
1. Demo: Press 'Demo' (w) GUI will generate a series of tones and demonstrate the CPT and VD



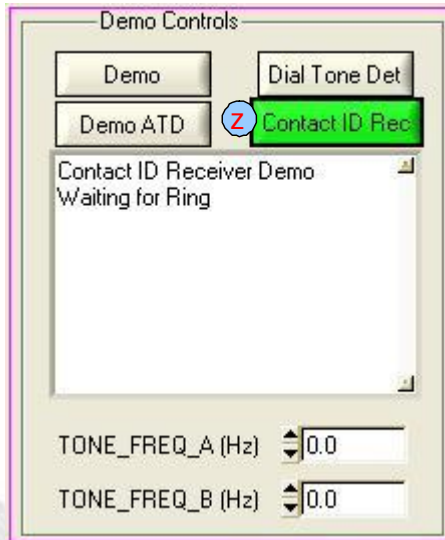
2. Demo ATD: Press 'Demo ATD' (x), GUI will generate a series of two tones and the CAS_STATUS (?) will show detected tones.



3. Dial Tone Det: Press 'Dial Tone Det' (y).



4. Contact ID Rec: Press 'Contact ID Rec' (z).



4. Evaluation Board Control

When the ISD61S00 EV board is connected to PC, three extra screens show up:

- EVB Control: Allows control of Evaluation board, such as floating the PCM and SPI buses for connection of external drivers or setting SPI speed, setting USB device sample rate.
- Audio Cmds: Controls the ISD61S00 play and record commands such as executing voice macros, erasing recorded messages etc.
- Digital Cmds: Controls the ISD61S00 digital commands such as downloading a project, reading back for verification etc.

4.1 Audio Cmds

ISD61S00 provides two message recording mechanisms:

- ‘Record Msg’ (A):
 - Record a message without specifying address. ISD61S00 automatically searches for available memory sectors starting from the RMP (Recording Memory Pointer) for the message recordings. When a ‘Stop’ command is issued to finish the recording, or the recording is stopped due to memory full, VPE automatically reads back the recorded message address and length.
- ‘Record Msg At’ (B):
 - Record a message at the user-specified address, which is reserved in advance. By default the reserved messages are allocated before the RMP. Since the addresses of Reserved Messages are allocated in advance, there is no need to read back the recorded message address. Reserved Message is especially useful for a predefined scenario.

The ‘Play Msg At’ command (C) plays a message at the user-specified address. Users can play a message from the middle by specifying an offset (D). One offset is one memory sector, which is about one second based on 8kHz 4-bit ADPCM. ISD61S00 does not provide a PAUSE command; however, a PAUSE command can be easily done by toggling between ‘Stop’ and ‘Play Msg At’ with an offset.

To erase a message, simply specify the address and then click on ‘Erase Msg At’ (E). Addresses of recorded messages are shown on the Message List. (F)

‘Get Msg List’ (G) is not an ISD61S00 command. It is indeed a bunch of digital-read commands used to scan the memory for messages. The first byte of each memory sector shows whether or not a beginning of message is inside the memory sector.

VM (H) & VP (I) sub-screens show users the VM/VP tags and indexes that they are working on.

To execute a VM or play a VP, simply click on the VM/VP sub-screens to get the index, and then click on ‘Execute VM’ (J) or ‘Play VP’ (K).

VM and VP are index based, which means to execute a VM or play a VP the microcontroller only sends a simple command followed by the index. It is address-free! Please note that VM and VP share the indexes, and it always starts from VM. For example, if a project has three VMs and four VPs, then the indexes 0 ~ 2 belong to VMs, and the indexes 3 ~ 6 belong to VPs.

‘Indirect Voice Prompts’ (L) is to make voice macro flexible. For example, it could be used to announce time and temperature in users’ system. To announce a sentence like ‘*It is 7:15 am; the outside temperature is 70 degree Fahrenheit*’, do the following:

- Microcontroller gets the time (**7:15 am**) and temperature (**70** degree Fahrenheit) from the system's timer and sensor.
- Write the corresponding voice prompt indexes of **7**, **15**, **am**, and **70** into R0, R1, R2, and R3.
- Execute the voice macro of that sentence.

To emulate it on VPE, click on VP sub-screen (**I**) to get the index, then right-click on Rn, n = 0 ~ 7, to update Rn, then click on 'Play VP@Rn' (**L**).

ISD61S00 provides four 'SPI Data Transfer' commands. For details, please refer to design guide. Take 'SPI Decode' (**M**) as an example, microcontroller sends compressed audio data (VPs or recorded messages) to ISD61S00 to decode. Double-click on '004 PLAY' (**N**) to set a playback path for 'SPI Decode'. Check 'CTS SPI' (**O**) if users want a continuous SPI transaction, which means VPE ignores all other tasks before the 'SPI Data Transfer' command finishes. Right click on the Filename blank (**P**) to assign a compressed audio file, and then click on 'SPI Decode' (**M**).

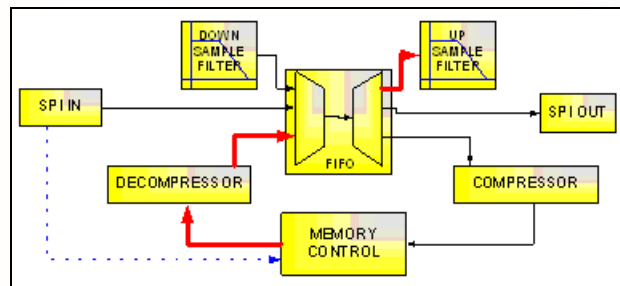


Figure 4-1 SPI Send Compressed Data to Decode

'Read ID' (**Q**) reads back the four-byte ISD61S00 ID.

'Read Status' and 'Read INT' (**R**) both read back the 4-byte ISD61S00 status. The only difference is that 'Read INT' clears the pending interrupt signal as well. ISD61S00 will not generate a new interrupt signal if there is a pending one.

'Saved Configurations' (**N**) is a copy from the 'Configuration' screen. Double-click on a setting to write to the ISD61S00.

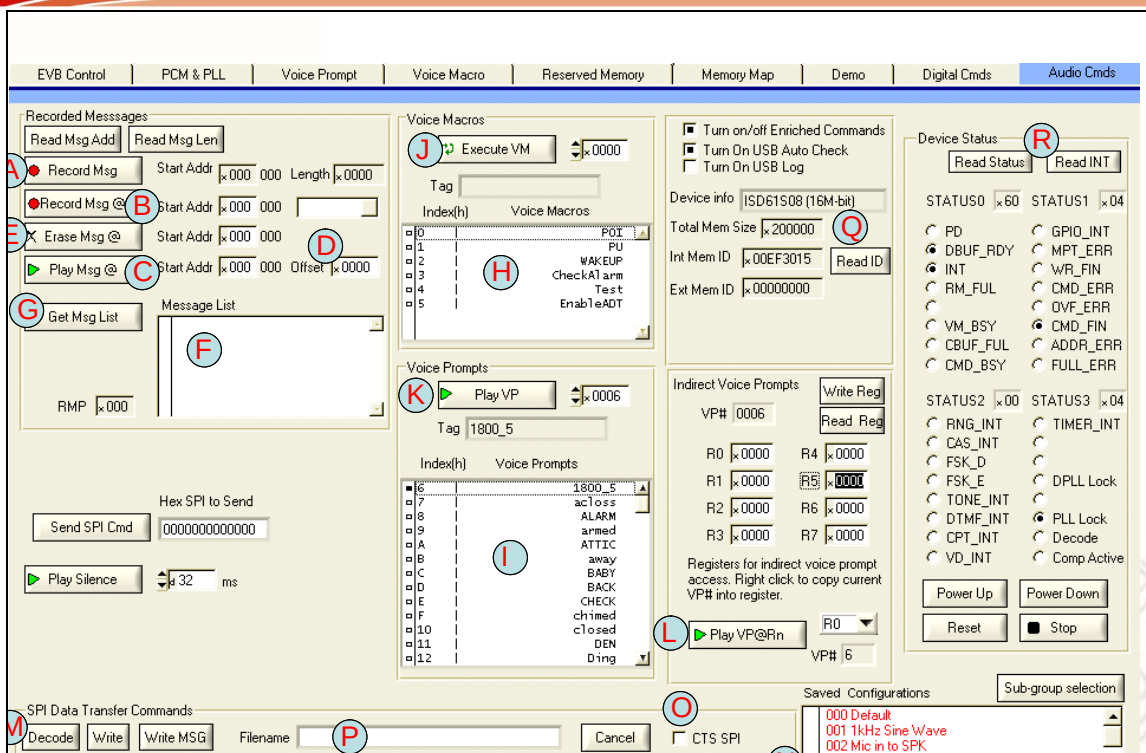


Figure 4-2 Audio Cmds

4.2 Digital Cmds

ISD61S00 provides four digital commands: digital-read (A), digital-write (B), erase-memory (C), and chip-erase (D).

To erase the whole memory, click on 'Chip Erase' (D); while the erasing is in progress, status bit CMD_BSY (E) goes high. Time for erasing the whole chip depends on the memory density. Below is a table for reference.

Device	Memory Density	Typical Erasing Time
ISD61S02	4Mbit	5 sec
ISD61S04	8Mbit	10 sec
ISD61S08	16Mbit	25 sec
ISD61S16	32Mbit	40 sec
ISD61S32	64Mbit	80 sec

Table 4-1 Chip Erasing Time

To erase partial memory, specify the start (F) and end addresses (or start address and number of sectors to erase), and then click on 'Erase Memory' (C); while the erasing is in progress, status bit CMD_BSY (E) goes high. The minimum erasable memory size is one sector (4k bytes).

Memory contents can be read back and either displayed on screen or saved as a file on PC.

To display the memory contents on screen:

- (3) Specify the start (G) and end addresses (or start address and number of bytes to read).

- (4) Specify to display on screen (**H**) (the default).
- (5) Click on 'Digital Read' (**A**).

To save the memory contents on PC:

- Specify the start (**G**) and end addresses (or start address and number of bytes to read).
 - Users can click on 'Read ID' (**I**) to get the memory size.
- Specify to save as a file (**H**). Click on 'Select File to Save' (**J**) to specify the path and filename.
- Click on 'Digital Read' (**A**).

Memory contents can be updated by digital-write from either a file on PC or from the screen.

To update the memory contents from a file:

1. Want to erase the whole chip and then download a file from PC:
 - Check 'Mass Erase/Prog/Reset' (**K**).
 - Specify to update from a file (**L**) (the default). Click on 'Select File to Open' (**M**) to specify the path and filename.
 - The start (**N**) and end addresses will be loaded automatically.
 - Click on 'Digital Write' (**B**), and VPE will chip-erase the whole memory and then download the file to the ISD61S00.
2. Just overwrite partial memory contents by a file from PC (the default):
 - Uncheck 'Mass Erase/Prog/Reset' (**K**).
 - Similar procedure as the above. VPE accomplishes this task via the help of PC:
 1. If needed,
 - Read back one 4kbyte sector and save on PC buffer.
 - Overwrite the buffer by the corresponding portion of the file.
 2. Sector-erase the sector.
 3. Download the buffer to the sector.
 4. Repeat the above steps.

To update the memory contents from the screen:

- Read out the ISD61S00 data and have it displayed on screen as described above.
- Edit the data contents on screen.
 - The first sector (sector 0) is about the ISD61S00 configuration. Do not mess it up.
- Specify to update the memory contents from screen (**L**).
- The start (**N**) and end addresses will be automatically matched.
- Click on 'Digital Write' (**B**); VPE accomplishes this task via the help of PC:
 1. If needed,
 - Read back one 4kbyte sector and save on PC buffer.
 - Overwrite the buffer by the corresponding portion of the file.
 2. Sector-erase the sector.
 3. Download the buffer to the sector.
 4. Repeat the above steps.

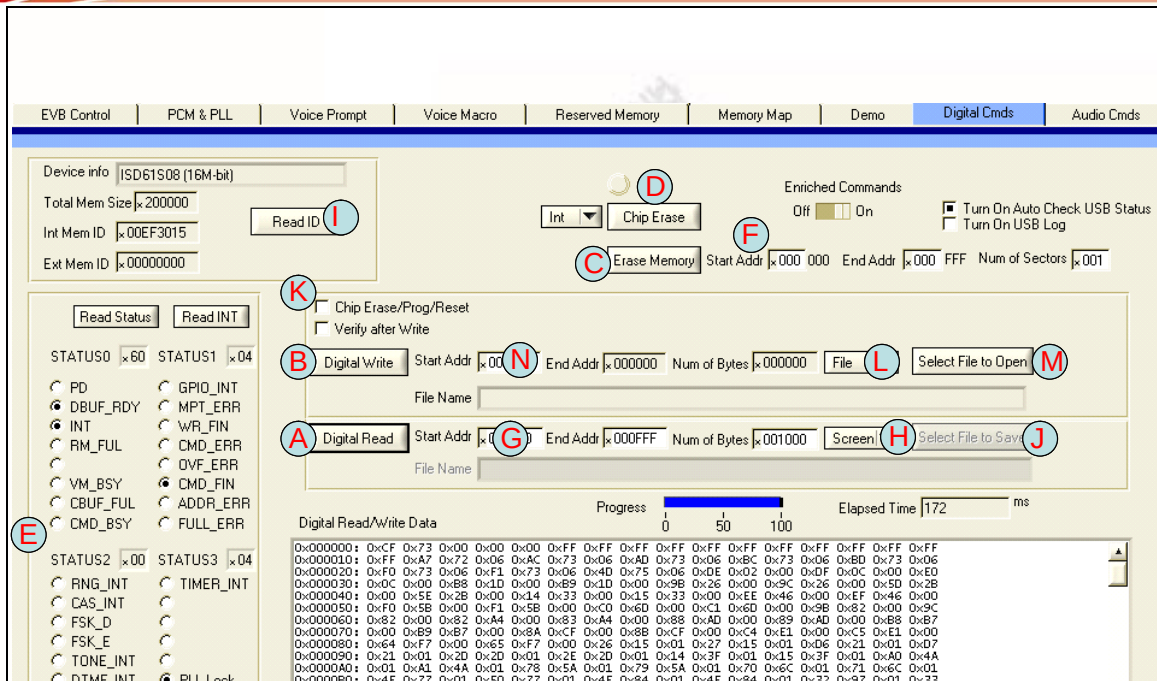
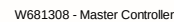
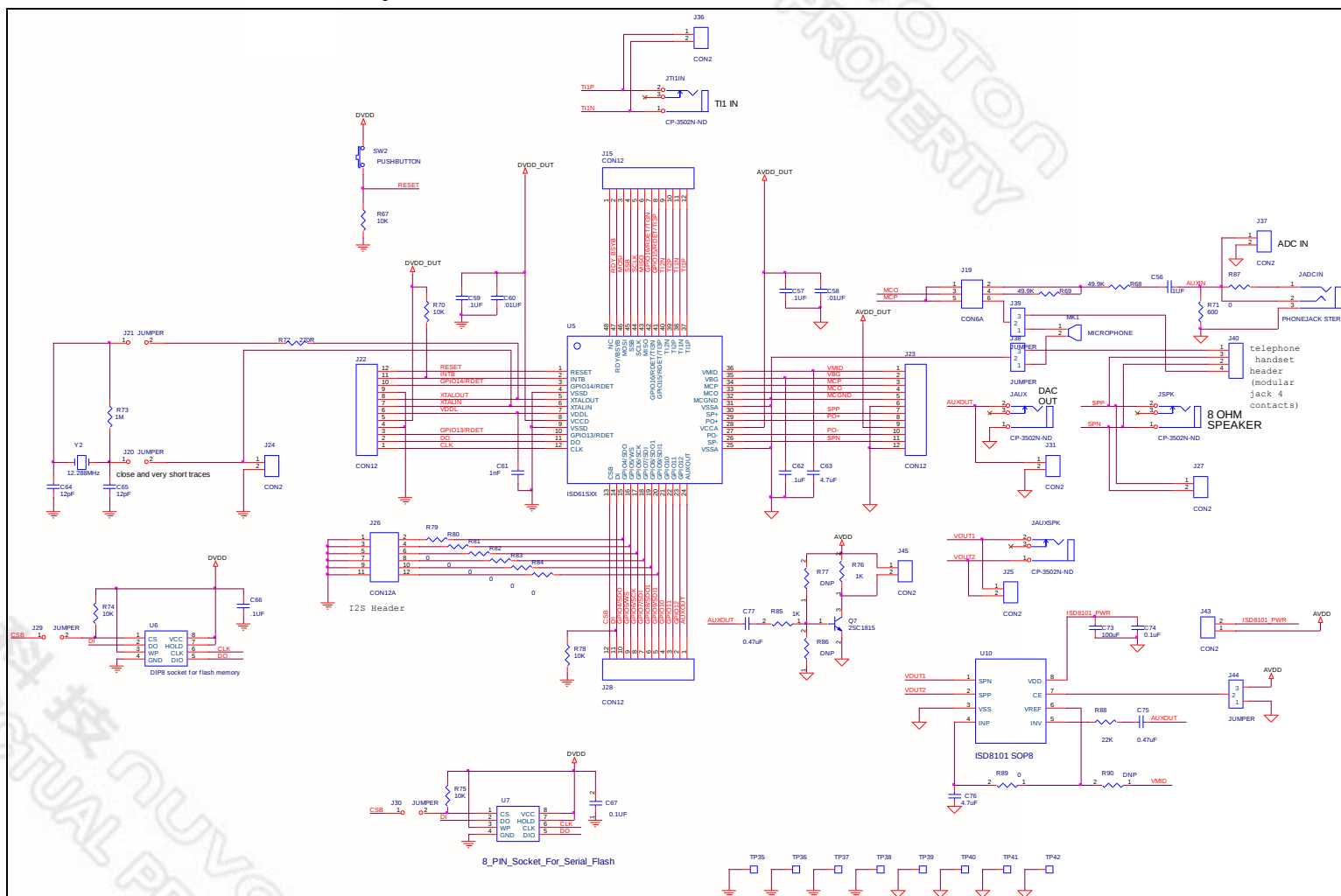


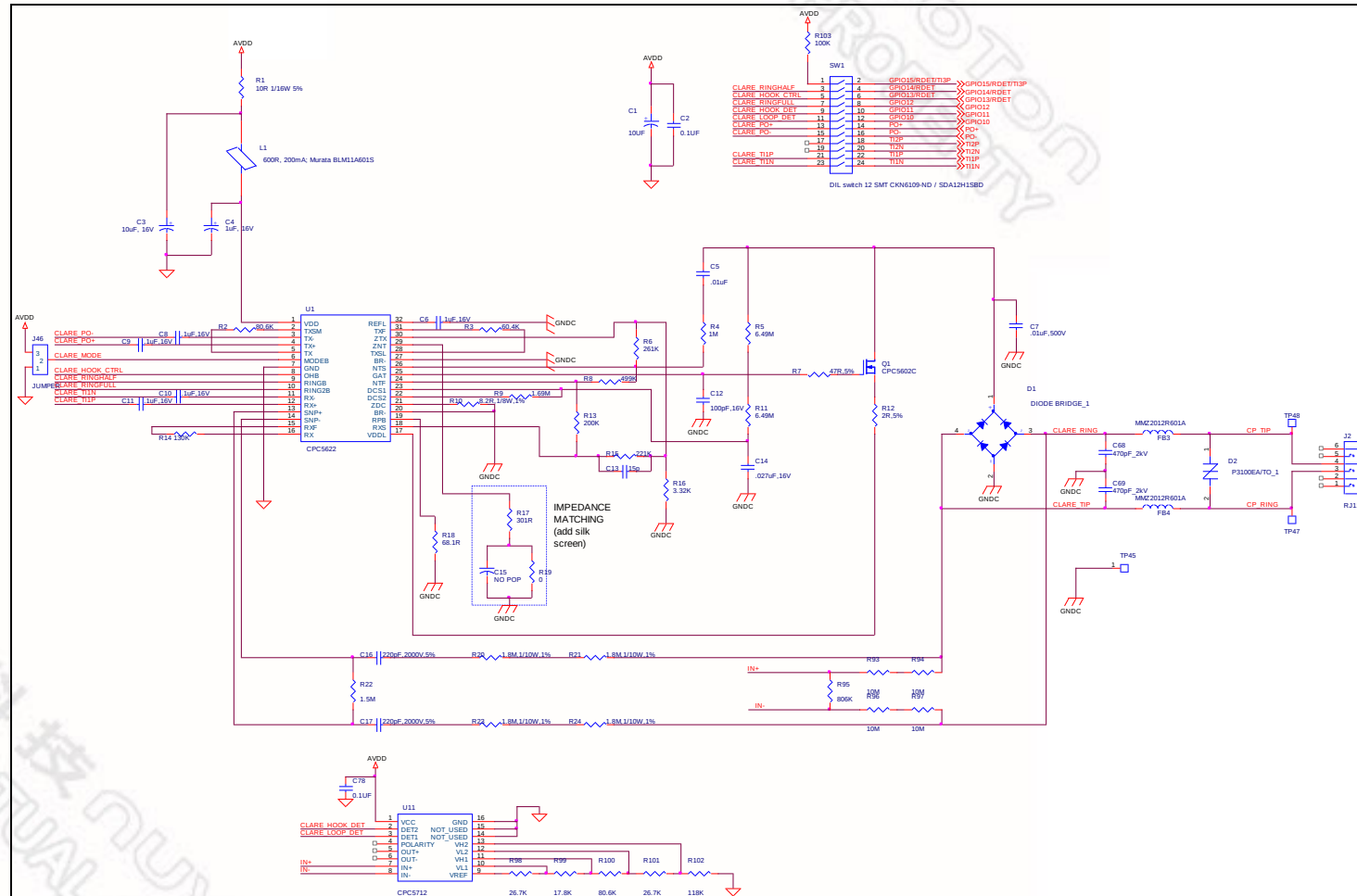
Figure 4-3 Digital Cmds

5.1 USB Controller and Power Sources

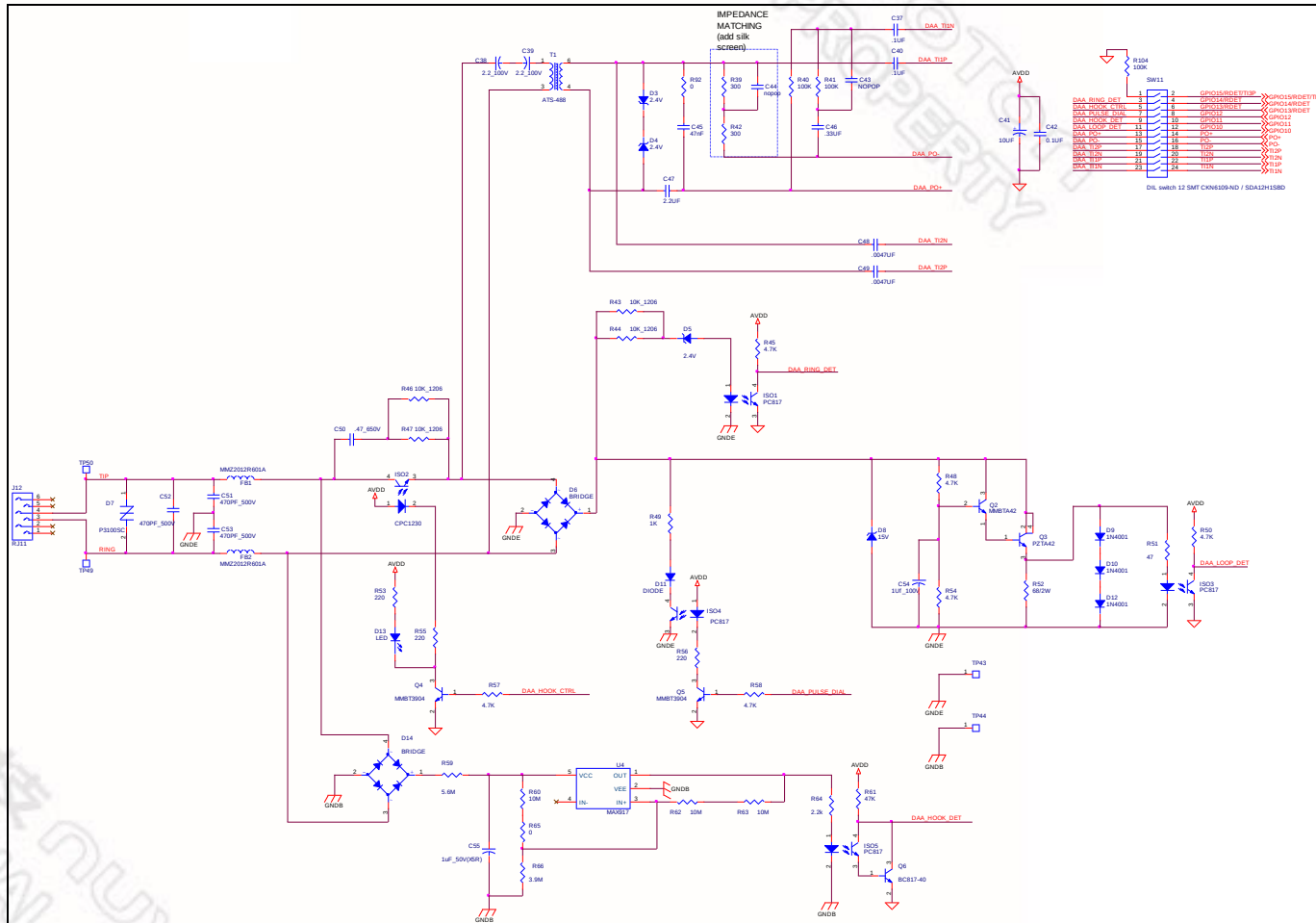


5.2 ISD61S00 and Related Circuitry





5.4 Discrete DAA



6. Revision History

Version	Date	Description
61.0010	Nov 10, 2008	Initial release:
61.0031	May 5, 2010	Added telephony description.